

LLM-Assisted Ontology Design for Restaurant Menu Data: a case study

Ali Amini^{1,*†}

¹*Department of Mathematics and Computer Science, Amirkabir University of Technology (Tehran Polytechnic), Tehran, Iran*

Abstract

Ontology design for domain-specific data, such as restaurant menus, is usually labor-intensive and vulnerable to semantic inconsistencies due to the complexity of modeling hierarchical relationships, semantics of category, and evolving data structures. In traditional ontology engineering methods, there is a huge need for human expertise to create classes, relationships, and vocabularies. In this paper, we look into the capabilities of Large Language Models (LLMs) to facilitate ontology engineering by automating the hierarchy extraction, identifying semantic relationships, and dynamic schema updates from semi-structured menu data. We use a small real-world dataset of menu items to do a quick test of the LLM's ability on four important points: (1) extracting hierarchical structures from CSV data, (2) recursively finding semantics and relationships, (3) suggesting schema changes if any data changes, and (4) making ontology development easier for people. In this research, our observation is reflecting the positive impact of LLMs on the speed of the whole process in Ontology Engineering with supervision and validation of human-in-the-loop. By presenting our observations here we aim to encourage more research in this area. In addition, we are continuing our work by creating plugins for ontology extraction AI agent which will be presented in our future researches.

Keywords

Large Language Model, Ontology, Knowledge Graph, Ontology Construction, Semantic Extraction Agent, Agentic Ontology System

1. Introduction

For many domains, Ontology Engineering was playing an important role in structuring and organizing the data. It enables users to run complex query, do reasoning and alignments between various data sources and systems. In specific domains like restaurant menu, ontology engineering requires the experts to understand the entities and their relationship among other entities like, categories, ingredients, menu definition, their prices and so on. Conventional ontology engineering methodologies, frequently implemented using tools such as Protégé, are characterized by their manual nature, requiring domain experts to manually establish classes, relationships, and controlled vocabularies. This labor-intensive process is prone to inconsistencies and semantic gaps, particularly in dynamic domains where data undergoes rapid evolution [1].

We recently observed huge progress in Large Language Models (LLMs), such as GPT-based systems. These advancements show abilities to understand and the structure of the data better in many aspects like extracting relationships, and generating meaningful knowledge schema [2]. All these NLP technology advancement will introduce potential to decrease the manual burden of ontology design by automating tasks such as hierarchy extraction, semantic relationship detection, and schema evolution. However, despite much interest in LLMs for knowledge graph construction, their specific role in ontology engineering for restaurant menu data remains underexplored [3].

In this short paper, we provide an initial exploration of LLM-assisted ontology design for restaurant menus. We lightly evaluate the feasibility of leveraging LLMs to address four aspects of ontology construction: (1) extracting hierarchical structures from flat menu data (e.g., mapping menu categories to items and ingredients), (2) detecting semantics recursively (e.g., identifying relationships across

2nd International Workshop on Symbolic and Generative AI for Science (SymGenAI4Sci), co-located with SEMANTiCS 2026: 22nd International Conference on Semantic Systems, September 15, 2026, Ghent, Belgium

✉ aliamini99@aut.ac.ir (A. Amini)

ORCID 0009-0006-4810-9197 (A. Amini)



© 2026 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

items and ingredient hierarchies), (3) proposing schema evolution in response to data changes, and (4) reducing the overall workload for human ontology engineers. Our goal is not to provide an exhaustive evaluation but to highlight key opportunities and challenges, laying the groundwork for future, more systematic research in this area.

2. Related Work

2.1. Traditional Ontology Engineering Methods

For years, ontology engineering was limited to using tools like Protégé by domain experts to model the data manually and extract the classes, properties, and axioms in a systematic manner [4]. Definitely the accuracy of these approaches is high because experts have high control over what they find in the data. However, it takes lots of time for humans to model large datasets and it is hard to re-visit the ontology every time the data changes, like restaurant menu systems. Because of these limitations facing the manual ontology engineering, the semi-automatic methods emerged and became more popular which are using NLP pipelines to extract concepts and relationships from both structured and unstructured data sources [5]. Tokenization, named entity recognition, and part-of-speech tagging are all parts of these pipelines, along with other similar processes. But these methods are usually only used in certain situations because they need to be set up and tuned for each domain. So it's making them weak and hard to use with more than one dataset.

2.2. AI and NLP in Ontology Learning

There are huge changes in the usage of artificial intelligence for ontology modeling application and these usage has changed from rule-based systems to statistical models and deep learning models. In ontology learning the traditional application of artificial intelligence was through the use of pattern mining techniques to extract relationships between terms using association rules and to extract concept hierarchies from large corpora [6]. However, more recent studies in neural ontology learning leverage embedding and sequence modeling to incorporate contextual relationships [7]. All these methods give more flexibility and ease of scalable architecture than traditional approaches; however, they want a lot of data for training and they are very domain specific. In addition, these approaches are performing good in extracting the concepts but they lack the ability in understanding the hierarchy and semantics behind the relationships. Despite this, the deployment of ontology engineering techniques is still developing and has many issues in accuracy, consistency and control of logical extraction, especially in applied topics like food and restaurants. On the other hand, transformer-based generative models (Large Language Models) including GPT-4, have shown a different approaches which is coming from their ability in language understanding and content creation. LLMs can find logical relationships and entities without following any rules and templates [8, 9]. However, the LLM-assisted approaches are very new to ontology engineering, and challenges related to accuracy, consistency, and management persist, especially in practical areas like food and hospitality[10].

3. LLM-driven Ontology Construction Framework

This section is about the dataset used, the design of the LLM-based approach for ontology construction, and the evaluation strategy used in this study. Our methodology is designed to provide a light evaluation of the feasibility of LLM-assisted ontology design, rather than a comprehensive benchmark.

3.1. Dataset And Usecase

We used the Restaurant Menu Items Profitability dataset available on Kaggle ¹, which contains structured information on restaurant menu items, including:

¹<https://www.kaggle.com/datasets/rabieelkharoua/predict-restaurant-menu-items-profitability>

It has 1000 records of data, in 4 categories of Appetizer, Beverage, Main Course, Desserts and 16 unique menu items as shown in Table 1. This is sample of rows but can show how the columns are related in this semi-structured dataset. The dataset provides a rich but flat structure, which is suitable for testing LLM-based inference of hierarchies (e.g., MenuCategory → MenuItem → Ingredient), semantic grouping, and schema evolution.

Table 1: Sample of Restaurant Menu Dataset

Rest.ID	Category	MenuItem	Ingredients
R003	Beverages	Soda	[confidential]
R001	Appetizers	Spinach Artichoke Dip	[Tomatoes, Basil, Garlic, Olive Oil]
R003	Desserts	New York Cheese-cake	[Chocolate, Butter, Sugar, Eggs]
R003	Main Course	Chicken Alfredo	[Chicken, Fettuccine, Alfredo Sauce, Parmesan]

- RestaurantID: Unique identifier of the restaurant (e.g., R001, R003).
- MenuCategory: High-level category (e.g., Appetizers, Beverages, Desserts, Main Course).
- MenuItem: Name of the menu item (e.g., Soda, Spinach Artichoke Dip, Chicken Alfredo).
- Ingredients: List of main ingredients for each menu item.
- Price: Price of the menu item.
- Profitability: Label indicating the profitability of the item (Low, Medium, High).

3.2. LLM-Assisted Ontology Generation

To generate ontology parts from CSV data like classes, relationships, and properties (attributes), we used GPT-4. Our prompting approach includes three iterative stages:

Hierarchy Extraction:

Input: A subset of rows from the dataset in CSV format.

Task: Ask the LLM to infer an ontology schema by identifying classes (e.g., Restaurant, MenuItem), hierarchical relationships (MenuCategory → MenuItem), and ingredient relationships.

Example Prompt: “From the following menu data, infer the class hierarchy and relationships to create a conceptual ontology (e.g., Restaurant offers MenuItem, MenuItem belongsTo MenuCategory).”

Thing

```

|-- Restaurant
|--   |-- RestaurantID (datatype property)
|-- MenuCategory
|--   |-- Beverages
|--   |-- Appetizers
|--   |-- ...
|-- MenuItem
|--   |-- name (datatype property)
|--   |-- price (datatype property)
|--   |-- profitability (datatype property)
|--   |-- belongsTo → MenuCategory
|--   |-- offeredBy → Restaurant
|--   |-- hasIngredient → Ingredient
|-- Ingredient
|--   |-- name (e.g., 'Garlic', 'Basil', ...)
```

Recursive Semantic Detection:

We prompted the LLM to recursively refine relationships, such as identifying ingredient hierarchies (e.g., “Parmesan” is a type of “Cheese”) or cuisine types. This recursive approach allowed the model

to generate multi-level semantic groupings beyond the flat structure of the CSV. For the recursion to happen, we can programmatically call the same prompt on each **rdfs:range** to the depth of our choice depending on the granularity level of the graph we want to build. Below is an example of calling LLM to extract the semantics (classes, properties) from the csv:

```
:Restaurant a owl:Class .
:MenuItem a owl:Class .
:MenuCategory a owl:Class .
:Ingredient a owl:Class .
:IngredientType a owl:Class .
:CuisineType a owl:Class .

:offers a owl:ObjectProperty ;
    rdfs:domain :Restaurant ;
    rdfs:range :MenuItem .

:belongsTo a owl:ObjectProperty ;
    rdfs:domain :MenuItem ;
    rdfs:range :MenuCategory .

:hasIngredients a owl:ObjectProperty ;
    rdfs:domain :MenuItem ;
    rdfs:range :Ingredient .

:partOfCuisine a owl:ObjectProperty ;
    rdfs:domain :MenuItem ;
    rdfs:range :CuisineType .

:isA a owl:ObjectProperty ;
    rdfs:domain :Ingredient ;
    rdfs:range :IngredientType .

:hasPrice a owl:DatatypeProperty ;
    rdfs:domain :MenuItem ;
    rdfs:range xsd:float .
```

Dynamic Schema Evolution: When new data entries are introduced—such as unseen categories ("Fusion Dishes", "Vegan Specials") or attributes (e.g., hasSpiciness)—the agent reevaluates the existing schema and prompts the LLM to adapt the ontology. It assesses whether to extend the current model or restructure certain branches as seen in Figure 2. Throughout this process, it maintains logical consistency by cross-checking with existing axioms and class/property domains.

3.3. Ontology Development AI Agent

In the traditional approach to ontology development using LLMs, the model is treated as a one-time schema generator: it takes a prompt and dataset as input and generates an ontology as output. This approach, however, does not take into consideration the fact that the interaction between the model and the user is not a one-time event but a dynamic process in real-world knowledge engineering, where ontologies must evolve with expanding data, shifting contexts, and growing domain complexity.

Our proposed ontology development agent views LLMs as collaborative and continuously improving entities, not as black-box text generators. The agent does not simply use the LLMs as a means of generating a one-time output, but remains engaged with them as a means of continuously improving the schema through three key abilities:

Memory: This agent has access to all prior interactions, concepts, and structural choices, which gives it the consistency in iterations. Having memory makes the agent to refer to its memory and so it avoids any repetition in its process by keeping the history of them and adds new elements without affecting the hierarchy.

Recursion: Recursion is happening by prompting the agent recursively to get the class structures and all relationships among them. As an illustration, following the extraction of a class like *Ingredient*, the agent can further explore its subtypes (e.g., *Herb*, *Dairy*, *Protein*) through the same inference mechanism. Consequently, this facilitates multi-level semantic enrichment, which develops organically as new instances or categories are integrated.

Conditional Refinement: By using such "rule-based" checks or "confidence" thresholds in the generation loop, the agent can validate the output and reprompt the LLM if it identifies any inconsistencies, ambiguities, or missing links. Thus, a feedback loop is created to improve the quality of the ontology.

In this way, the development of the ontology is no longer a static process; rather, it is a dynamic one through the above model of agents. The agent thus facilitates the integration between the automated process and the expert process to refine the ontology. This model doesn't only make the usability of LLMs with semantic systems easier but also facilitates the usage of the above model with other systems such as enterprise knowledge graphs and digital twins. Figure 1 is showing how the interaction between the user and the Ontology Design Agent looks like.

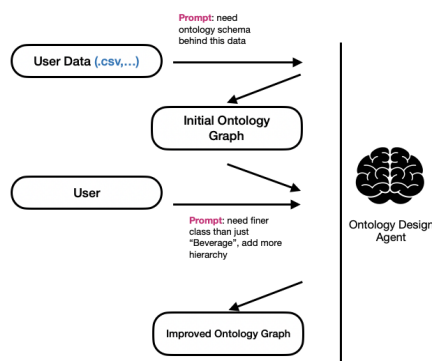


Figure 1: Ontology Design Agent System Overview

3.4. Evaluation Strategy

Given the exploratory nature of this work, we focused on light, qualitative and quantitative evaluation:

Qualitative Assessment: A manual check of the hierarchies and semantic relationships generated by the LLM was conducted to ensure the semantic consistency of the generated hierarchies. The quality of the generated hierarchies with compare to the domain was also ensured. For example, the relationships between the categories of the menu and the corresponding items (e.g., mapping "Cheesecake" under "Desserts" rather than "Main Course") were checked. Overall, the quality of the generated schema and inferred relationships was found to be of a high standard, showing the coherence with domain expectations and low semantic errors.

Quantitative Assessment: We looked assessed precision and recall to see how accurate the LLM was at the TBox level. The resulting class relationships were compared to a small, manually created ground truth ontology containing 100 menu items. This evaluation was possible because the extracted hierarchical relationships were broken down into individual triples, similar to subject-predicate-object statements. For example, "Tiramisu → hasCategoryOf → Dessert." A random sample of 100 triples was chosen from this collection for manual review. In this particular cohort, a substantial 92% of the triples demonstrated both accuracy and semantic relevance. The primary objective of this investigation was to assess the degree to which the identified conceptual relationships aligned with the data instances present within the ABox.

3.5. Limitations of Evaluation

The current evaluation is intentionally limited to facilitate comprehensive investigation. Initially, the assessment is predicated on a modest subset of the dataset (small sample). Secondly, a general, pre-trained LLM was employed, devoid of fine-tuning or domain-specific adjustments; this may have influenced the model’s capacity to find specific semantic differences that are only related to the restaurant sector. Ultimately, the primary objective of this study is to show the feasibility and pinpoint avenues for future research, rather than to provide exhaustive benchmarking or a thorough quantitative analysis.

4. Challenges and Limitations

Semantic Challenges Semantic extraction from menu data encounters a substantial hurdle in the form of abstract semantics, where the exact meaning of a term cannot be definitively established through simple linguistic approaches such as substring matching or keyword-based rules. Traditional natural language processing (NLP) techniques often rely on heuristic rules; for example, they might deduce that "Cheeseburger" is a subtype of "Burger" based on shared words. However, such approaches doesn’t apply in more nuanced cases. For instance, while “Philly Cheesesteak” contains the word “steak,” it is not grouped as a type of steak but rather as a type of sandwich. Because of the lack of understanding the context in rule-based approaches, they misclassify such subtle differences in the vocabulary. On the other hand, LLM performs well in understanding the context (due to learning on huge corpora) and because of this ability they can capture these semantics better. In our observation, we noticed that LLMs can better identify and classify the menu item complexities rather than inferring only from lexical signals.

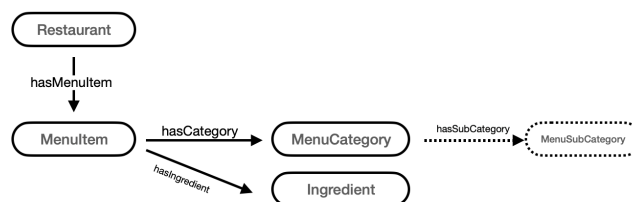


Figure 2: Proposed Ontology Schema by GPT-4

5. Conclusion and Future Work

In this paper, we explored the feasibility of using Large Language Models (LLMs) to assist with ontology design in the context of restaurant menu data. Through qualitative and quantitative assessment, we showed that LLMs are capable of extracting meaningful class hierarchies, detecting recursive semantic relationships, and suggesting schema evolutions from semi-structured tabular input. The ability to produce logically consistent and semantically meaningful ontologies with high precision in relations without requiring fine-tuning makes LLMs show promise as tools for developing ontologies beyond rule-based methods.

As future work, we continue working on a more comprehensive research including larger and more diverse menu datasets, integration with ontology alignment techniques, and fine-tuning LLMs on domain-specific data to improve accuracy and domain sensitivity. Additionally, including reasoning tools (e.g., OWL reasoners) and adding question-based evaluations could provide a more rigorous understanding of logical completeness and usability.

In this regard, as an extension of this research, we are currently working on developing a plugin for an ontology extraction agent. This plugin will be designed to automate schema generation from semi-structured data in a modular and reusable manner. It will be introduced in future work as a single contribution, further advancing the practical applications of LLM-integrated ontology engineering.

Declaration on Generative AI

The authors used generative AI tools (ChatGPT 4.o) during the preparation of this manuscript, primarily for language refinement and grammar and spelling checks. All technical content, analysis, and conclusions were developed and verified by the authors. The authors take full responsibility for the content of this paper.

References

- [1] M. C. Suárez-Figueroa, R. García-Castro, B. Villazón-Terrazas, A. Gómez-Pérez, Essentials in ontology engineering: Methodologies, languages, and tools (2011).
- [2] Z. Bi, J. Chen, Y. Jiang, F. Xiong, W. Guo, H. Chen, N. Zhang, Codekgc: Code language model for generative knowledge graph construction, *ACM Transactions on Asian and Low-Resource Language Information Processing* 23 (2024) 1–16.
- [3] R. Amini, S. S. Norouzi, P. Hitzler, R. Amini, Towards complex ontology alignment using large language models, in: *International Knowledge Graph and Semantic Web Conference*, Springer, 2024, pp. 17–31.
- [4] N. F. Noy, D. L. McGuinness, et al., *Ontology development 101: A guide to creating your first ontology*, 2001.
- [5] A. Maedche, S. Staab, *Ontology learning for the semantic web*, *IEEE Intelligent systems* 16 (2005) 72–79.
- [6] P. Cimiano, *Ontology learning and population from text: algorithms, evaluation and applications*, Springer, 2006.
- [7] D. Barina, T. Chlubna, M. Solony, D. Dlabaja, P. Zemcik, Evaluation of 4d light field compression methods, *arXiv preprint arXiv:1905.07432* (2019).
- [8] A. S. Lippolis, M. J. Saeedizade, R. Keskiä, S. Zuppiroli, M. Ceriani, A. Gangemi, E. Blomqvist, A. G. Nuzzolese, *Ontology generation using large language models*, in: *European semantic web conference*, Springer, 2025, pp. 321–341.
- [9] P. Salovskii, I. Gorshkova, Automatic ontology construction using llms as an external layer of memory, verification, and planning for hybrid intelligent systems, *arXiv preprint arXiv:2604.20795* (2026).
- [10] A. Amini, An agentic framework for dynamic ontology generation, in: *LLM-Integrated Knowledge Graph Generation From Text (TEXT2KG) and International BiKE Challenge 2026*, volume 4233, *CEUR Workshop Proceedings*, 2026, pp. 128–137.