Guiding LLM Generated Mappings with Lifecycle-Based Metadata: An Early Evaluation

Sarah Alzahrani[1,2,*], Declan O'Sullivan[3]

[1]School of Computer Science and Statistics, Trinity College Dublin, Ireland
[2] Imam Mohammad Ibn Saud Islamic University (IMSIU), Saudi Arabia
[3]ADAPT Center, Trinity College Dublin, Ireland

Abstract

Large Language Models (LLMs) are increasingly used to automate knowledge engineering tasks such as generating RDF mappings. While promising, their outputs often lack semantic precision, syntactic correctness, and contextual metadata. This paper investigates whether structured metadata aligned with the mapping lifecycle can improve the quality and reusability of LLM-generated mappings. We present a metadata model that covers key phases of the mapping process and integrate it into the MetaSEMAP tool to support context-aware prompting. Using real-world uplift scenarios, we compare RML outputs generated from unguided prompts with those informed by lifecycle metadata. Our initial findings show that guided prompts consistently produce syntactically valid, semantically rich, and FAIR-aligned mappings. These results highlight the potential of structured metadata to guide LLMs toward generating higher-quality and reusable semantic artifacts in knowledge graph construction.

Keywords

Metadata, Declarative mappings, LLMs, Mapping lifecycle, Knowledge graphs, Context engineering

1. Introduction

Declarative mappings like RML[1] are essential for transforming structured data into RDF for use in knowledge graphs. However, creating such mappings is a technically demanding task, often requiring expertise in both syntax and domain-specific ontologies [1]. Recently, large language models (LLMs) have shown promise in assisting with this task by generating knowledge graphs and mappings from natural language descriptions [2, 3, 4]. Despite their capabilities, LLMs often struggle with producing outputs that are semantically accurate, syntactically valid, and compliant with established standards [5], and human-in-the-loop validation is often recommended to ensure quality and reliability.

This paper explores whether incorporating structured metadata into LLM prompts can improve the quality of generated mappings. This approach aligns with recent work in context engineering, which focuses on enriching prompts with structured and task-specific context to enhance the reliability and relevance of LLM outputs [6]. We focus on a real-world scenario involving the creation of mappings to uplift data from Ireland's open data portal[2].and demonstrate how metadata guidance, based on a mapping lifecycle model, influences the outputs generated by gpt-3.5-turbo (via OpenAI API).

The remainder of this paper is structured as follows: we review related work on declarative mappings, metadata standards, and the role of LLMs in semantic data generation. We then introduce our lifecycle-based metadata model and its integration into the MetaSEMAP prompting interface. Next, we present the experiment design and results comparing guided and unguided prompt outputs. Finally, we conclude the paper and outline future research directions.

The Second Workshop on Knowledge Graphs and Neurosymbolic AI (KG-NeSy), co-located with SEMANTiCS'25: International Conference on Semantic Systems, September 3–5, 2025, Vienna, Austria
salzahra@tcd.ie (S. Alzahrani); declan.osullivan@tcd.ie (D. O'Sullivan)
https://www.scss.tcd.ie/Declan.OSullivan/ (D. O'Sullivan)
0000-0003-2169-7006 (S. Alzahrani); 0000-0001-7770-2117 (D. O'Sullivan)

[1]RML (RDF Mapping Language) is a declarative language for mapping heterogeneous data to RDF. See: http://rml.io/spec.html
[2]Ireland's open data portal: https://data.gov.ie

2. Background and Related Work

2.1. Declarative Mappings and Metadata Need

Declarative mapping languages such as R2RML and its extension RML enable structured transformation of heterogeneous data sources into RDF for use in knowledge graphs [1]. These languages define how data is semantically uplifted without requiring custom code. RML, in particular, supports various input formats like CSV, JSON, and XML using the RML vocabulary in combination with query languages such as XPath and JSONPath. Despite its expressiveness, authoring RML mappings remains a technically demanding task. Tools like YARRRML [7] and documentation frameworks such as RMLDoc [8] aim to simplify this process by offering user-friendly syntaxes or structured documentation. However, the lack of embedded metadata to describe the purpose, provenance, and context of mappings limits their interpretability and reuse.

2.2. Metadata Standards for Mappings

To enhance transparency and reusability, several metadata standards have been proposed. General-purpose vocabularies like Dublin Core and DCAT provide terms for describing datasets and publishing context. More specialized efforts, such as SSSOM [9], focus on metadata for ontology alignments, while MQV [10] supports the assessment of mapping quality. Recent initiatives like FAIR-IMPACT [11] proposed metadata and best practices for describing mappings as FAIR Digital Objects, aiming to improve their findability, accessibility, and interoperability. These efforts offer valuable guidance on core metadata elements such as provenance, mapping purpose, and versioning, but remain largely conceptual and do not yet support a comprehensive, lifecycle-based framework. As a result, a unified metadata model that cover the full mapping lifecycle including design, generation, documentation, and reuse is still lacking. To address this, we propose a structured lifecycle-based metadata model that captures key elements across five phases of mapping activities. This model, introduced and evaluated in our earlier work [12, 13], is further detailed later in the paper.

2.3. Large Language Models in Semantic Mapping

Large Language Models (LLMs), such as GPT-3 and GPT-4, have demonstrated the ability to generate RDF triples, SPARQL queries, and declarative mappings such as RML from natural language inputs [3]. This capability presents new opportunities for simplifying the knowledge graph construction process, particularly for non-experts. However, while promising, the outputs of LLMs often lack semantic precision and completeness. Studies have shown that LLM-generated mappings frequently omit critical components such as input declarations, namespace prefixes, and join conditions, and they rarely include metadata supporting provenance, reuse, or quality assessment [14, 3, 2]. Moreover, these mappings may include hallucinated classes or properties that do not exist in the target ontology, making them unreliable for use.

To address these challenges, we propose a structured metadata-driven approach that guides LLMs using context aligned with the mapping lifecycle. By injecting metadata such as the mapping's purpose, data source characteristics, design decisions, and provenance, we provide the model with domain-specific constraints and intentions that help shape the generated outputs. This approach improves the accuracy, interpretability, and reusability of the resulting mappings by grounding generation in explicit and verifiable context.

This aligns with emerging practices in context engineering [6], where structured metadata and formal vocabularies help constrain and contextualize LLM behavior to produce more accurate and consistent outputs. In our case, the metadata model acts as a source of prompt structure, supporting reproducibility and enhancing trust in LLM-assisted mapping generation.

3. Lifecycle Metadata Model

To support high-quality and reusable semantic mappings, we proposed a lifecycle-based metadata model that organizes metadata across five key phases: analysis, design, development, testing, and maintenance. This model was first introduced in our earlier work [12] and later validated through a community study with Semantic Web practitioners [13]. It is implemented in the MetaMap tool [15], enabling structured metadata capture through a guided interface.

The model includes fields for describing the mapping purpose, input sources, stakeholders, design decisions, tooling, validation outcomes, and publishing context. Table 1 presents a summary of the key metadata fields grouped by lifecycle phase. The full specification is publicly available on GitHub [16].

In this paper, we extend the model's use beyond documentation by incorporating selected fields into LLM prompts. This approach evaluates whether structured metadata can guide LLMs to generate more accurate, standards-compliant, and reusable RML mappings.

Table 1
Proposed Metadata Fields Aligned with Mapping Lifecycle Phases

Lifecycle Phase	Key Metadata Fields
Analysis	Stakeholder details (URI, name, role, organization); mapping purpose (requirements, domain, assumptions, risks); input description (URI, name, source, type, creator, format)
Design	Final design decisions, justifications, expected quality metrics
Development	Mapping process metadata (mapping URI, tools used, method, algorithm, start/end date, format)
Testing	Testing metadata (type, timestamp, test results)
Maintenance	Versioning and publishing information (publisher name, source, version number, version date)

4. Methodology

4.1. Experimental Setup

To evaluate whether lifecycle-based metadata improves the quality of LLM-generated RML mappings, we designed an experiment comparing two prompting strategies: (1) unguided prompts with only a task description, and (2) guided prompts augmented with structured metadata. We used `gpt-3.5-turbo` via the OpenAI Python SDK v1.0 (`chat.completions.create`) to generate RML outputs. All prompts were submitted through a controlled interface (Figure 1) to ensure consistent model behavior and prompt formatting across both guided and unguided conditions.

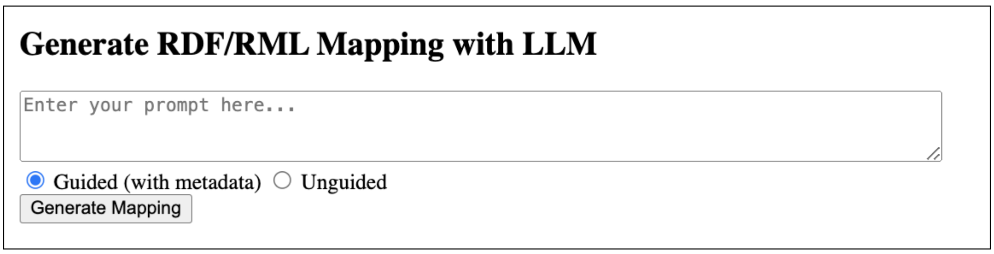

Figure 1: The interface to generate RML outputs.

4.2. Mapping Scenarios

We evaluate our approach using three real-world scenarios derived from publicly available datasets on Ireland's open data portal[2]. Full dataset links and mapping files are available in our GitHub repository[3]. In Scenario 1 (S1), we use the *Counties, National Statutory Boundaries, 2019* dataset to convert a CSV file containing administrative boundary data for Irish counties into RDF. Although the dataset includes multiple attributes such as object IDs, area size, Irish names, and shape geometry, only a relevant subset was selected for the purpose of the experiment.

In Scenario 2 (S2), we use the *G0421, Population per NUTS 3 Region* dataset to convert a JSON file containing population data for Irish regions into RDF. Although the dataset includes various statistical fields such as region codes, units of measurement, and percentage values, only a relevant subset was selected for the purpose of the experiment. The mapping focused on region identifiers, population counts, and temporal attributes to construct RDF triples that reflect regional population distributions in a given year.

In Scenario 3 (S3), we use the *CSO Electoral Divisions, National Statistical Boundaries, 2022* dataset to convert a CSV file containing administrative division data for electoral districts in Ireland into RDF. Although the dataset includes various attributes such as object IDs, Irish and English names, shape geometry, and codes from multiple authorities, only a relevant subset was selected for the purpose of the experiment. The mapping focused on modeling spatial hierarchy and containment relationships. Each electoral division was assigned a hierarchical URI based on its `ED_ID` and linked to its parent county and province using a custom property `ex:containedIn`.

4.3. Prompt Design

To compare the effects of metadata guidance, each mapping scenario was submitted to gpt-3.5-turbo using two types of prompts: *unguided* and *guided*. The unguided prompt included only a task description, while the guided prompt was enriched with structured metadata drawn from our lifecycle-based model, which defines 37 fields across five phases. We selected 17 fields most relevant to prompt-based mapping generation, focusing on descriptors from the *Analysis* and *Design* phases (e.g., *Purpose*, *Mapping Type*, *Mapping Domain*, *Input Description*, *Final Design Decisions*, *Justification*, and *Quality Metrics*). We also included contextual publishing and output fields from the *Maintenance* phase such as *Publisher Source*, *Version Number*, *Version Date*, *Output Format*, and *Output Syntax*, which clarify expected structure and encourage standards compliance. These fields were chosen based on their impact on design-time decisions and feedback from previous user studies [13] showing their relevance to mapping quality and reuse.

Fields from the *Development* and *Testing* phases (e.g., *Tools*, *Mapping Algorithm*, *Testing Result*) were excluded as they relate to post-generation steps. Similarly, fields with limited influence on generative quality (e.g., *Risks*, *Stakeholder Background*) were omitted. The selected metadata was embedded in natural language form to simulate structured context engineering and guide the LLM in producing syntactically valid, semantically rich, and reusable RML mappings. Table 2 shows an example of both prompt types used in Scenario 1 (S1). The full prompt sets for Scenarios 2 and 3 are available in our GitHub repository[3].

5. Evaluation

Each mapping was assessed based on three key dimensions. *Correctness* refers to whether the output is syntactically valid RML and can be parsed without errors. *Structure awareness* evaluates how well the mapping captures the structure of the input data, including the correct use of `rml:logicalSource`, `rml:referenceFormulation`, and iterators. *Semantic quality* considers whether the mapping uses meaningful classes, properties, and URIs aligned with domain semantics and whether it includes relevant metadata to support reuse. Across all three scenarios, guided prompts produced mappings that

[3]Project repository: https://github.com/sarah-alzahrani/LLM.

Table 2
Example of Guided and Unguided Prompt for Scenario 1

Prompt Type	Prompt Text
Unguided	Generate an RML mapping to convert a CSV file of Irish county boundaries into RDF. Each row contains a county ID, a county name, and geometry coordinates.
Guided	Generate an RML mapping to convert a CSV file of Irish county boundaries into RDF. Each row contains a county ID, a county name, and geometry coordinates. **Metadata:** **Stakeholder Name:** Ordnance Survey Ireland, **Role:** Data provider **Purpose:** Publish administrative boundaries as linked geodata **Mapping Type:** CSV to RDF (RML) **Mapping Domain:** Geospatial / Administrative **Input Description:** CSV file with county ID, names, and geometry **Input Name:** counties.csv , **Input Format:** CSV **Final Design Decisions:** Use `rdfs:label`, `schema:latitude`, `schema:longitude`, and `schema:addressRegion` to represent the attributes. **Justification:** Align with schema.org and LinkedGeoData best practices **Quality Metrics:** All 32 counties covered, valid spatial attributes **Publisher Source:** https://data.gov.ie/dataset/counties-national-statutory-boundaries-20191 **Version Number:** 1.0 , **Version Date/Time:** 2025-06-04 **Output Format:** Turtle (.ttl), **Output Syntax:** RML

were consistently more correct, structurally aware, and semantically richer than those from unguided prompts. In particular, all guided prompts correctly used `rml:logicalSource`, specifying input type and reference formulation (e.g., `ql:CSV` or `ql:JSONPath`), and included `iterator` when handling JSON inputs. In contrast, unguided prompts often defaulted to `rr:logicalTable`, which is valid R2RML syntax but insufficient for non-tabular or nested data. This distinction is important because `logicalSource` is the required mechanism in RML for handling diverse data sources, and its absence leads to incorrect or incomplete mappings, especially for formats like JSON. For example:

- **Scenario 1 (Counties – CSV)**: The unguided mapping used `rr:logicalTable`, ignoring CSV-specific reference formulation. The guided version used `rml:logicalSource` with `ql:CSV`, along with appropriate classes and coordinate properties.
- **Scenario 2 (Population – JSON)**: The unguided prompt omitted both the `iterator` and JSON path references, resulting in an unusable mapping. The guided prompt correctly used `ql:JSONPath`, specified the `iterator`, and aligned the schema with population data standards.
- **Scenario 3 (Electoral Divisions – CSV)**: The unguided output lacked hierarchical URI structure and containment logic. The guided version constructed meaningful URIs using region and division identifiers, and modeled geographic relationships with custom and standard vocabularies.

These results highlight how metadata-enriched prompting helps LLMs handle different mapping complexities by injecting contextual knowledge into generation. Guided prompts also produced reusable outputs that included metadata blocks aligned with FAIR and provenance principles, which were entirely missing from unguided versions. Overall, our initial evaluation shows that context-aware prompting grounded in lifecycle metadata improves both the technical correctness and semantic value of LLM-generated mappings, supporting better reuse, validation, and documentation. While this early evaluation offers qualitative insights, future work will expand the analysis with quantitative scoring, completeness metrics, and broader use case coverage to more systematically validate the approach.

6. Conclusion and Future Work

This study investigated the effect of lifecycle-based metadata guidance on the quality of RML mappings generated by LLM. By comparing guided and unguided prompts, we observed that metadata-enriched prompting significantly improves the syntactic accuracy, semantic richness, and standards compliance of LLM-generated mappings. Guided prompts led to the consistent use of appropriate vocabularies, correct handling of input formats, and the inclusion of metadata blocks that enhance provenance and reuse. These initial findings support a context-aware approach that combines the generative flexibility of LLMs with structured metadata to improve the reliability, interpretability, and reusability of semantic outputs. In future work, we plan to extend this evaluation framework to additional mapping types, compare performance across different LLMs, as well as investigate whether prompting LLMs to generate declarative mappings (like RML) results in more reusable outputs than directly generating RDF triples. Our initial findings highlight the value of structured metadata not just for documentation but as a guide in prompt-based knowledge graph construction. As LLMs become embedded in semantic workflows, the community must consider not only what outputs they generate but also how those outputs are guided, contextualized, and made reusable for others.

7. Acknowledgments

The first author acknowledges financial support from Al Imam Mohammad Ibn Saud Islamic University and the Saudi government — represented by the Royal Embassy of Saudi Arabia — Cultural Bureau in Dublin. The second author is partially supported by the SFI ADAPT Research Centre (grant number 13/RC/2106_P2).

Declaration on Generative AI

During the preparation of this work, the author used ChatGPT (GPT-4) and Grammarly for the purposes of grammar improvements. All AI-generated content was thoroughly reviewed, edited, and validated by the author, who takes full responsibility for the final manuscript and all its content.

References

[1] A. Dimou, M. V. Sande, P. Colpaert, E. Mannens, R. Verborgh, R. V. de Walle, Rml: A generic language for integrated rdf mappings of heterogeneous data, in: Proceedings of the 7th Workshop on Linked Data on the Web (LDOW), 2014.

[2] M. Freund, R. Dorsch, S. Schmid, A. Harth, Mapping by example: Towards an rml mapping reverse engineering pipeline, in: ESWC 2025 Workshop on Knowledge Graph Construction (KGCW), 2025.

[3] M. Hofer, J. Frey, E. Rahm, Towards self-configuring knowledge graph construction pipelines using llms – a case study with rml, in: Proceedings of the 6th International Workshop on Knowledge Graph Construction (KGCW@ESWC), 2024.

[4] B. Zhang, H. Soh, Extract, define, canonicalize: An llm-based framework for knowledge graph construction, arXiv preprint arXiv:2404.03868 (2024). URL: https://arxiv.org/abs/2404.03868. doi:10.48550/arXiv.2404.03868, proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing.

[5] V. K. Kommineni, B. König-Ries, S. Samuel, From human experts to machines: An llm supported approach to ontology and knowledge graph construction, arXiv preprint arXiv:2402.11155 (2024). URL: https://arxiv.org/abs/2403.08345.

[6] A. Volkov, What is context engineering?, 2024. URL: https://blog.getzep.com/what-is-context-engineering/.

[7] P. Heyvaert, A. Dimou, R. Verborgh, Yarrrml: Yet another rml rendering language, in: Proceedings of the 1st International Workshop on Knowledge Graph Construction (KGC 2018), volume 2160 of *CEUR Workshop Proceedings*, 2018.

[8] P. Heyvaert, A. Dimou, R. Verborgh, Rmldoc: Generating documentation for declarative mapping rules, in: Proceedings of the 22nd International Semantic Web Conference (ISWC 2023), volume 14265 of *Lecture Notes in Computer Science*, Springer, 2023, pp. 169–185. doi:10.1007/978-3-031-41393-4_11.

[9] N. Matentzoglu, et al., A simple standard for sharing ontological mappings (sssom), Database 2022 (2022) baac035. doi:10.1093/database/baac035.

[10] A. Randles, A. C. Junior, D. O'Sullivan, A vocabulary for describing mapping quality assessment, refinement and validation, in: 2021 IEEE 15th International Conference on Semantic Computing (ICSC), 2021, pp. 425–430. doi:10.1109/ICSC50631.2021.00076.

[11] Y. Le Franc, N. Grau, N. Juty, G. Mejias, P. Reed, P. Ramezani, M. Poveda-Villalón, D. Garijo, C. Goble, R. van Horik, D4.5 - Guidelines and methodology to create, document and share mappings and crosswalks (V1.1), Technical Report, FAIR-IMPACT, 2025. URL: https://doi.org/10.5281/zenodo.15111167. doi:10.5281/zenodo.15111167.

[12] S. Alzahrani, D. Sullivan, Towards a unified metadata model for semantic and data mappings, in: Ontology Matching workshop at ISWC 2022, 2022.

[13] S. Alzahrani, D. O'Sullivan, What metadata is needed for semantic and data mappings?, in: 2023 IEEE 17th International Conference on Semantic Computing (ICSC), 2023, pp. 312–316. doi:10.1109/ICSC56153.2023.00064.

[14] M. Parciak, B. Vandevoort, F. Neven, L. Peeters, S. Vansummeren, Schema matching with large language models: An experimental study, in: Proceedings of the VLDB 2024 Workshops, CEUR-WS, 2024.

[15] S. Alzahrani, D. O'Sullivan, Metamap: Facilitating mapping reuse and lifecycle documentation, in: 2024 IEEE 18th International Conference on Semantic Computing (ICSC), 2024, pp. 323–328. doi:10.1109/ICSC59802.2024.00058.

[16] S. Alzahrani, Metasemap metadata model, 2025. URL: https://github.com/SarahAlzahranitcd/MetaSEMAP-Metadata.