Systematic Evaluation of Rule-Based Analytics for LLM-Driven Graph Data Modelling

Fabio Yáñez-Romero
University Institute for Computer Research
University of Alicante
fabio.yanez@ua.es

Andres Montoyo
Armando Suárez
Department of Computing and Information Systems
University of Alicante
montoyo@dlsi.ua.es
armando@dlsi.ua.es

Alejandro Piad-Morffis
Yudivian Almeida-Cruz
School of Math and Computer Science
University of Havana
apiad@matcom.uh.cu
yudivian@matcom.uh.cu

Abstract

Artificial intelligence models have increasingly supplanted traditional rule-based systems for extracting knowledge from structured data; however, the integration of both approaches remains underexplored. While large language models offer greater flexibility than rigid rule systems, the structured knowledge from rule-based analytics can significantly enhance LLM performance and efficiency. This paper presents a novel multi-agent system that automatically generates graph database schemas from tabular data by strategically combining rule-based analytics with large language models. Our system utilises a lightweight rule framework that selects the most suitable analytical methods based on column data types, providing targeted insights to inform the schema generation process. The system's modular architecture enables comprehensive ablation studies examining both the effectiveness of rule-based analytics and their optimal presentation formats. Through systematic evaluation, we demonstrate that structured rule formats reduce result variability (lower standard deviation) while contextualised formats achieve superior performance despite higher variance. Our analysis identifies which pipeline stages benefit most from analytical guidance, providing insights for optimising hybrid AI systems. This work contributes a practical framework for integrating rule-based knowledge with modern language models, demonstrating measurable improvements in both consistency and performance for structured data processing tasks.

1 Introduction

The evolution of natural language processing has involved different rule-based (Miller et al., 1996), statistical (Weikum, 2002), and machine learning systems (Galanis et al., 2021), culminating in the current dominance of Large Language Models (LLMs) (Feng et al., 2025). However, recent approaches suggest that there is room for improvement with techniques traditionally used in rule-based systems when combined with LLMs (Laqrichi, 2024). While LLMs have revolutionised most NLP tasks with their exceptional reasoning capabilities, they still face challenges with complex linguistic phenomena, scalability, and domain-specific accuracy requirements (Gururaja et al., 2023). These limitations have revived interest in knowledge-based and rule-based approaches, which offer superior explainability and remain competitive in niche domains (Chen et al., 2025).

Rule-based analytics have been the cornerstone of classical information extraction from structured data (Atzmüller et al., 2008), involving the extraction of entities, properties, and relationships via discovered data types. However, these analytical methods, while interpretable and precise, lack the semantic interpretability necessary to accurately handle multi-column relationships and implied patterns.

Contemporary causal language models demonstrate a remarkable capacity to understand structured data formats such as CSV, JSON, and Markdown (Oh et al., 2025), enabling them to reason over tabular data when provided with appropriate context. For automatic generation of graph database schemas from relational ones, such a combination is particularly valuable. Relational databases represent entities as tables with primary keys and associated columns, and relationships as foreign keys. Although this structure guarantees coherence and integrity, it is not suitable for tasks involving the detection of implicit relationships, hierarchical understanding, or semantic flexibility—the essential ingredients for graph-based representations.

Our approach demonstrates how rule-based analytics can be integrated systematically with LLMs to address these challenges. We employ a rule-

Proceedings of the First Workshop on Comparative Performance Evaluation: From Rules to Language Models associated with RANLP 2025, pages 154–164, Varna, Bulgaria, Sep 11, 2025.
https://doi.org/10.26615/978-954-452-102-8-016

based system that infers data types for every column and calls specialised analytical routines based on these type determinations. These analytics are then exposed as structured or contextualised context to LLMs in a multi-agent system, allowing us to contrast the relative performance impact of rule-based preprocessing on LLM-based schema generation.

The multi-agent system architecture enables systematic ablation studies by selectively masking analytical components, allowing us to quantify the contribution of rule-based analytics to overall system performance. Each agent specialises in different aspects: individual table analysis leveraging type-specific rules, cross-table relationship detection, and schema standardisation and integration.

2 Related Work

The automatic generation of graph database schemas from relational data represents a convergence of several fundamental research areas. Our work builds upon three interconnected domains: semantic interpretation techniques for extracting meaning from relational data, methodologies for converting relational schemas to graph representations, and the integration of large language models with tabular data processing.

2.1 Semantic Interpretation in Relational Data

The interpretation of semantics in tabular data has evolved significantly from early rule-based systems and heuristics (Cremaschi et al., 2024) to machine learning approaches (Chen et al., 2019). Traditional approaches relied primarily on unsupervised clustering techniques and supervised learning methods for column type classification and entity disambiguation. The introduction of dense vector representations marked a paradigm shift (Gorishniy et al., 2023), with specialised embedding techniques designed for tabular data enabling effective representation of column semantics, entity relationships, and cross-table linkages.

The emergence of large language models has fundamentally transformed semantic interpretation by enabling contextualised understanding of table content and structure (Cremaschi et al., 2025). Encoder-only models, such as BERT, have demonstrated effectiveness for header classification and column similarity assessment (Trabelsi et al., 2022). In contrast, decoder-only models such as Llama (Jiang et al., 2024) excel at entity linking, relationship extraction, and cross-table reasoning through in-context learning.

2.2 From Relational to Graph Databases

The conversion from relational to graph database schemas represents a critical challenge in modern data management (Bhandari and Chitrakar, 2024). While relational databases ensure data integrity through rigid schemas with primary keys, foreign keys, and predefined relationships, their structural constraints limit adaptability for downstream tasks requiring flexible semantic modelling.

Graph databases address these limitations by representing entities as nodes and relationships as edges, enabling more flexible modelling of semantic relationships. The conversion process involves identifying entities (potentially distributed across multiple tables), detecting implicit semantic relationships, and standardising properties and types. This transformation requires careful consideration of graph type selection (property graphs vs. RDF), structural properties (directionality, multigraphs), and higher-level semantic rules (Putrama and Martinek, 2022).

The complexity of this conversion process has motivated researchers to explore automated approaches leveraging advanced reasoning capabilities, leading to increased interest in utilising large language models for schema conversion (Sui et al., 2024a).

2.3 LLMs Integration with Tabular Data

Large Language Models have demonstrated remarkable capabilities in processing structured data through advanced prompt engineering techniques such as Chain-of-Thought reasoning (Wang et al., 2024) and in-context learning (Wen et al., 2025). However, several critical limitations constrain their effectiveness:

Format Sensitivity: LLMs exhibit pronounced sensitivity to tabular serialisation methods, with performance degradation of approximately 50% when tables are transposed (Liu et al., 2023). HTML and XML formats demonstrate superior performance with GPT models (Sui et al., 2024a).

Context Window Limitations: Context constraints pose significant challenges when processing larger tables, leading to performance degradation and the "lost-in-the-middle" phenomenon (Sui et al., 2024b).

Reliability Concerns: LLM outputs remain prone to hallucinations (Su et al., 2024), particularly in sensitive applications, with severity increasing as output length extends (Harrington et al., 2024). Mitigation strategies include audit modules and self-correction mechanisms (Karbasi et al., 2025).

External Tool Integration: The integration of external tools has significantly enhanced LLM utility for tabular data tasks, enabling code generation for database interaction (Zhang et al., 2023) and automated data processing workflows (Fan et al., 2024).

Despite these advances, current approaches primarily rely on commercial LLMs (Chen et al., 2025), limiting reproducibility and raising privacy concerns. Furthermore, existing methods lack a systematic evaluation of how rule-based analytics can enhance LLM performance in schema generation tasks, representing a significant gap that our work addresses.

3 MultiAgent System

To systematically evaluate the impact of rule-based analytics on graph schema generation from tabular data, we developed a multi-agent system that integrates data analytics with causal language models. Our primary objective is to generate valid graph database schemas from relational tabular data while enabling controlled experimentation to assess the contribution of rule-based preprocessing to overall system performance.

3.1 System Architecture and Design Principles

We implemented our system using LangGraph (Wang and Duan, 2024), a framework that enables the definition of distinct state graphs for different processing pipelines. This architectural choice provides crucial flexibility for our experimental design, allowing us to conduct ablation studies by selectively turning on or off specific nodes and analytics-driven prompts within the language model workflows. This modular approach facilitates systematic comparison between schema generation with and without rule-based analytical enhancement.

Our system architecture mirrors the decision-making process employed by expert graph database modellers when converting relational databases to graph representations (De Virgilio et al., 2013). The design incorporates domain expertise through a structured two-stage approach that addresses the inherent complexity of semantic interpretation and schema transformation. A comprehensive diagram illustrating the state graph used in our experiments, with and without analytics integration, is presented in Figure 1.

3.2 Processing Pipeline Architecture

The schema generation process operates through two complementary stages designed to capture both intra-table and cross-table semantic relationships:

1. **Table-Based Processing Pipeline**: This stage executes individual state graphs for each table in the source dataset, focusing on entity identification, relationship discovery, and property mapping within the context of each isolated table.

2. **Cross-Table Processing Pipeline**: This stage utilises a unified state graph to standardise redundant entities and relationships across tables, while identifying cross-table relationships, including primary and foreign key associations.

This dual-stage approach enables a systematic evaluation of how rule-based analytics influence various aspects of the schema generation process, ranging from local entity recognition to global schema coherence.

3.3 Table-Based Processing Pipeline

The table-level state graph implements three sequential processing nodes, each designed to leverage rule-based analytics for enhanced semantic understanding:

1. **Entity Identification**: Our system infers one or multiple entities within individual tables or recognises tables that lack sufficient information for entity extraction. When no entities are identified by the language model for a specific table, the table is excluded from the current pipeline stage.

2. **Intra-Table Relationship Discovery**: When multiple entities are detected within the same table, the language model infers relationships between those entities.

3. **Property Mapping**: For each column in a table, the system calls the language model to associate the column with identified entities or

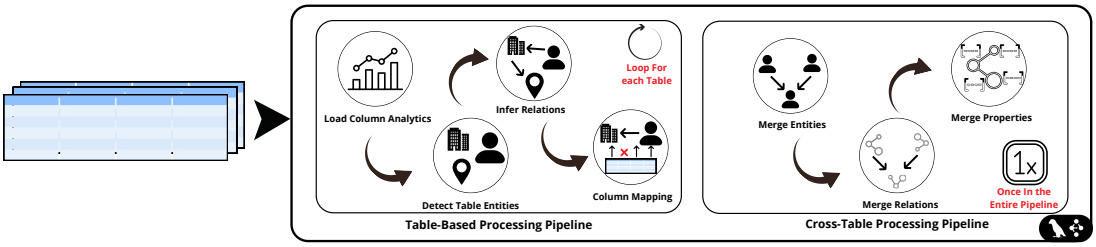

Figure 1: Entire Architecture for the system. Each table is processed individually before merging entities, relations and properties.

relationships. This process can be enhanced by providing the analytics related to that specific column.

The underlying strategy leverages the language model's ability to identify entities and relationships based on primary and foreign key analysis, enriched by rule-based analytics that provide deeper insights into column semantics and value distributions.

3.4 Cross-Table Processing Pipeline

The cross-table state graph operates on aggregated context from all processed tables to ensure schema consistency and completeness:

1. **Entity Standardisation**: The language model examines all previously identified entities, considering their names and associated properties through the initial columns and determine which semantically equivalent entities should be merged.

2. **Relationship Standardisation**: This process is activated when merged entities possess relationships with different names but equivalent semantic meanings. The model assigns the most appropriate name to these semantically equivalent relationships, ensuring schema coherence and reducing redundancy.

3. **Property Standardisation**: After entity and relationship standardisation, the system validates that all properties from merged components are correctly preserved and consolidated. The module identifies potential property conflicts arising from merging (such as duplicate properties with different data types) and applies resolution strategies to maintain schema integrity. This validation step is crucial for preserving the semantic richness captured during the table-based processing phase.

This systematic approach enables a precise evaluation of how rule-based analytics contribute to various aspects of schema generation, ranging from local semantic interpretation to global schema standardisation and consistency. The code for using the agent, as well as reproducing the entire experiment, can be found on GitHub[1].

4 Experimental Settings

Building upon the multi-agent system architecture described in the previous section, we designed a comprehensive experimental framework to systematically evaluate the impact of rule-based analytics on graph schema generation performance. Our experimental design enables controlled ablation studies that isolate the contribution of different analytical approaches to the overall effectiveness of the system.

4.1 Rule-Based Analytics Integration

The core hypothesis of our work centres on the premise that rule-based analytics can significantly enhance LLM performance in semantic interpretation tasks. To test this hypothesis, we implemented a type-specific analytical system that applies tailored analytics based on automatically inferred column data types. Our rule-based system categorises columns into four fundamental data types: categorical (including Boolean), string, numerical (including integer and float values) and date. The selection of specific analytics for each data type is grounded in established data science practices that optimise information extraction based on the inherent characteristics of each data type.

A detailed specification of the analytics performed for each data type is presented in Figure 2. These analytics range from basic statistical measures (mean, variance, distribution characteristics) to samples and automatically generated descrip-

[1] Repository for the Agentic Framework

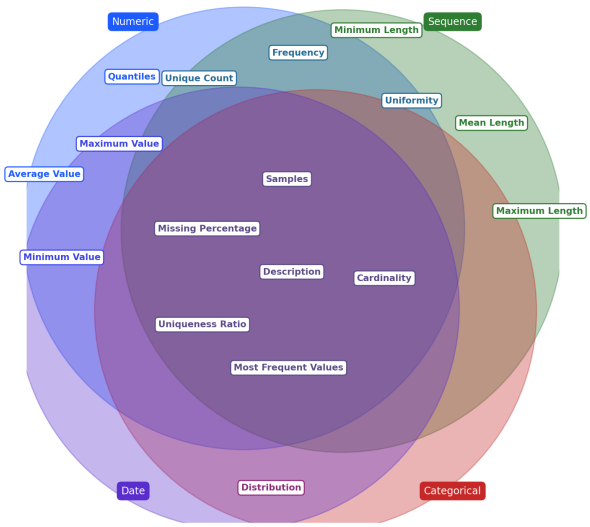

Figure 2: Analytics performed according to each data type detected. The intersections in the Venn Diagram represent the analytics that are shared among different data types.

tions of the entire columns, providing rich information for LLM decision-making.

4.2 Experimental Configurations

To systematically evaluate the contribution of rule-based analytics, we designed three distinct experimental configurations that represent different levels of analytical integration:

1. **Version 1 "No Analytics Baseline"** (V_1): This configuration serves as our baseline, providing only representative data examples for each column without any analytical context. This experiment enables direct measurement of the analytical contribution by comparing performance against pure LLM reasoning capabilities.

2. **Version 2 "Structured Analytic"** (V_2): This configuration provides comprehensive analytical results in a structured JSON format, exactly as computed and stored by our rule-based system. This approach tests the capability of the language model for understanding structural information while maintaining a clear organisational structure that facilitates systematic processing.

3. **Version 3 "Contextualised Analytic"** (V_3): This configuration applies analytical contextualisation methodologies inspired by successful approaches such as DeepJoin (Dong et al., 2023), which demonstrated significant improvements in semantic table interpretation through effective context integration. In this version, raw analytical results are transformed into natural language descriptions that provide semantic context about column characteristics, distributions, and relationships.

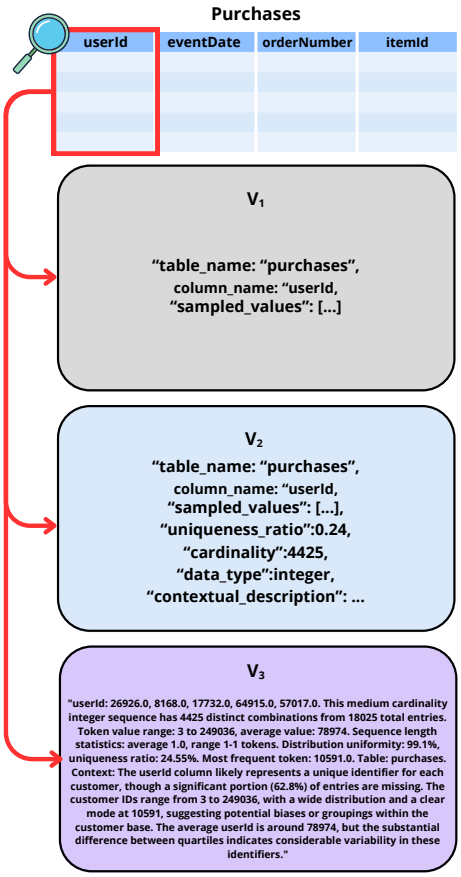

Figure 3: Analytics context formats supplied to the LLM-based agent for inferring the graph (property-graph) schema from the tabular dataset. V_2 encodes the analytics as structured JSON, while V_3 ("contextualised analytics") expresses the same information as narrative text generated deterministically from V_2 via a Python function to mimic typical LLM prompts. The figure contrasts these formats to assess how structured versus free-text context affects schema generation.

Importantly, all experimental variations utilise identical pipeline logic and differ only in the initial prompts provided to the language models. This design ensures that observed performance differences can be explicitly attributed to the presence or absence of rule-based analytics rather than architectural variations. A sample of each version is shown in Figure 3.

4.3 Implementation and Reproducibility Measures

To ensure experimental reproducibility and address the limitations of commercial LLM dependency identified in related work (Chen et al., 2025), we implemented our entire system using locally executed models. We selected Gemma 3 12B (Team et al., 2025) quantised to 4 bits, specifically the version hosted by Ollama[2], which provides an optimal balance between model capability and computational accessibility on standard user GPUs.

Our experimental configuration employs several measures to ensure reproducible results:

- Fixed Random Seed: All experiments use identical random seeds to ensure a consistent model behaviour across runs.
- Zero Temperature: Model temperature is set to zero to minimise stochastic variations in output generation.
- Model Consistency: The same model instance is used across all pipeline stages within each experimental run.
- Local Execution: All models are loaded and executed locally, eliminating external dependencies and ensuring data privacy.

Even with temperature = 0 and a fixed seed, LLM inference is not strictly deterministic: GPU-level numerical effects (e.g., parallel reductions, fused kernels, library autotuning) and decoder tie-breaking near probability ties can flip early tokens or the stopping point across runs (Atil et al., 2025), (Song et al., 2024). In our multi-agent pipeline, such micro-differences are amplified because each agent conditions on previous generations. We hypothesise that variability in answer length at early stages is the dominant driver: slightly longer/shorter completions change what downstream agents read, steering different trajectories and yielding different schema proposals. To address this, we ran 10 independent trials and report the mean and variance across runs.

4.4 Prompt Engineering Strategy

Our prompt design incorporates established techniques that have demonstrated effectiveness in structured data reasoning tasks. Specifically, we employ in-context learning examples that illustrate the desired schema generation behaviour, combined with Chain-of-Thought (CoT) reasoning prompts that guide the model through systematic analysis steps. This approach has proven particularly effective in interpreting tabular data, as demonstrated in recent literature (Liu et al., 2025). The complete prompt specifications for each experimental configuration are detailed in the experiment repository[3], enabling full reproducibility of our experimental setup. Each prompt variant maintains an identical logical structure while varying only in the analytical context provided to the language model.

4.5 Statistical Validation

To ensure the statistical significance of our results, each experimental configuration is executed ten times under identical conditions. We calculate both mean performance metrics and variance measures for each version of the experiment and for each dataset, enabling robust statistical analysis of the analytical contribution. This approach addresses the inherent variability in LLM outputs while providing sufficient statistical power to detect meaningful performance differences between analytical and baseline configurations. This experimental design directly addresses the research gap identified in our literature review regarding the systematic evaluation of rule-based analytics in LLM-driven schema generation tasks, providing a rigorous framework for assessing the effectiveness of our integrated approach.

4.6 Dataset

For our experimentation, we employ the Diginetica dataset, a large-scale benchmark released initially for the CIKM Cup 2016[4]. This dataset has become a cornerstone in session-based recommender system research due to its comprehensive coverage of real-world e-commerce interactions. Crucially for our purposes, the Diginetica dataset is organised into multiple interrelated tables, making it especially suitable for exploring the transition from a tabular to a graph-based data model:

- **Items:** Each product is uniquely identified and annotated with descriptive features such as price and textual tokens.

[2] Gemma 3 12B quantised model on Ollama

[3] Prompt Templates used in the experiments

[4] Original challenge where Diginetica Dataset was released

- **Categories:** Products are mapped to one or more categories, introducing a hierarchical structure that enriches the context for each item.
- **Views:** Every user interaction with a product page is captured, including session identifiers, temporal ordering, and user context.
- **Purchases:** Purchase events are linked to sessions and users, with references to related Items and Views, effectively connecting user actions across the dataset.
- **Queries:** This table logs user search activities with timestamps and contextual information, referencing entities from the other tables and enabling the reconstruction of full user search journeys.

The high degree of correlation and reference among these tables naturally aligns with the principles of graph data modelling, where entities (e.g., users, items, categories) become nodes and their relationships (such as views, purchases, and category memberships) are represented as edges. Such a structure facilitates the explicit modelling of complex interdependencies and interaction patterns that may be cumbersome to express or query efficiently in a purely tabular schema.

Therefore, Diginetica's rich, interconnected tabular design provides an ideal foundation for our task of translating traditional relational data into a graph database schema, enabling more expressive analysis and supporting advanced graph-based recommendation and user modelling techniques.

5 Results and Discussion

5.1 Evalutation Method

From the tabular dataset, we derived a lossless, agnostic property graph schema using Grok-4 (xAI, 2025). A graph data expert then reviewed and refined the naming, cardinalities, and data types to establish the expert-validated golden schema. We evaluated each experimental variant against this reference by measuring completeness (recall) over nodes, relationships, and properties; node/edge matching was synonym- and alias-aware to handle LLM naming variance, while property names were matched precisely to the original columns.

The completeness assessment methodology varied according to the schema component being evaluated:

- **Node completeness**: Measured by comparing the types of nodes present in the generated schema against those defined in the golden schema
- **Property completeness**: Assessed by determining whether nodes and relationships contain the properties they should possess based on the original relational database columns
- **Relationship completeness**: Evaluated based on whether relationships between existing node types match those in the golden schema, regardless of relationship names or directionality

The relationship evaluation methodology was deliberately simplified due to practical constraints. Language models frequently infer relationships with inverse orientations, incorrect directionality, or overly generic names. This complexity made the automatic evaluation of relationship completeness challenging and hindered the assessment of improvements in relationship detection across experimental versions.

5.2 Discussion of Results

The experimental results, presented in Table 1, show average outcomes and standard deviations across 10 independent tests per experimental version, along with the best-performing results for each version. Based on these findings, we can conclude the impact of column analytics usage and format on schema generation performance. The discussion is organised into specific component-level results and overall schema prediction performance.

5.2.1 Specific Results

Node Detection Performance: Node completeness showed minimal sensitivity to the use of analytics. When analytics were applied, unstructured sentence-format analytics proved counterproductive, with some contextualised analytics experiments degrading node type detection performance compared to baseline conditions.

Property Detection Performance: Property completeness, which depends solely on mapping columns to predefined entities, demonstrated a clear improvement with the use of analytics. Contextualised analytics format achieved the highest success rates in this component, suggesting that rich contextual information aids in accurate property-entity mapping.

Table 1: Completeness Percentage for Node, properties and relations, comparing the schema generated with the golden schema for Diginetica Dataset.

Completeness	**No Analytics** (V_1)	**Structured Analytics** (V_2)	**Contextualised Analytics** (V_3)
Node	$\mathbf{85.70 \pm 0.00}$	$\mathbf{85.70 \pm 0.00}$	82.84 ± 5.72
Property	70.87 ± 1.86	73.88 ± 3.74	$\mathbf{74.26 \pm 5.68}$
Relation	68.75 ± 13.98	63.75 ± 3.75	$\mathbf{75.00 \pm 11.18}$
Overall	75.11 ± 4.91	74.44 ± 1.11	$\mathbf{77.39 \pm 7.24}$

Relationship Detection Performance: Relationship completeness yielded mixed results across experimental conditions. Experiments without column analysis outperformed those using structured analytics, but underperformed compared to contextualised analytics approaches. This suggests a non-linear relationship between analytics complexity and the accuracy of relationship detection.

5.2.2 Overall Results

Overall, the best predictions were obtained using contextualised analysis (V_3), while the worst results were obtained using structured analytics. From the point of view of variability in results, the most uniform results are achieved between experiments using this set of structured analytics (V_2). In contrast, the most unpredictable results are obtained when the analytics are contextualised.

6 Discussion and Conclusion

The results indicate that while basic data analytics (providing representative column subsets along with column and table names) do not enhance node detection in inferred graph databases, they significantly improve property and relationship detection. Contextualised analytics demonstrated improvements of up to 9% in these components, with the format of contextual data proving critical for optimal relationship detection.

When evaluating overall schema generation effectiveness, contextualised analytics maximised model performance, while structured analytics yielded the poorest results. This suggests that rich, contextual information enables more accurate schema inference than rigid, structured data formats.

From a consistency perspective, structured analytics dramatically reduced result variability, as evidenced by lower standard deviations. This finding suggests that structured analytics should be preferred when result stability is prioritised over peak performance. Conversely, contextualised analytics produced the highest variability—exceeding even baseline conditions without analytics—making them the least stable approach across all experimental versions.

These findings present a clear trade-off between performance and stability in graph schema generation. Users prioritising maximum accuracy should employ contextualised analytics, despite increased result variability, while those requiring consistent, predictable outcomes may benefit from structured analytics approaches, albeit with reduced peak performance.

7 Limitations and Future Work

The experiments conducted present several limitations that we intend to address in future work, such as the use of open-source models of different sizes to verify the degradation/improvement based on model size.

Likewise, it would be of great interest to make a comparison with large commercial models, according to similar methodologies applied by previous works (Chen et al., 2025), which would give us an idea of what percentage of success can be expected with a multi-agent system like this compared to frontier models, being able to measure at this point also the computational cost associated with numerous calls of medium-sized models compared to the use of these commercial models.

On the other hand, the rule system used is extremely simple, with considerable room for improvement that can affect the final accuracy of the schema when determining the entities, relationships, and properties of the graph database.

Finally, truly understanding the limitations and capabilities of this system requires the use of more tabular data in various domains and with diverse characteristics, such as a large number of columns per table or tabular data that does not conform to the nomenclature of a relational database. In this sense, other structured formats provided for the analytics might be impactful on the final results, which needs further investigation.

Acknowledgments

This research has been funded by the University of Alicante, the Spanish Ministry of Science and Innovation, the Generalitat Valenciana, the Valencian Agency for Innovation (AVI), and the European Regional Development Fund (ERDF) through the following funding:"GeoIA: Artificial GeoIntelligence platform to solve citizens problems and facilitate strategic decision making in public administrations" (INNEST/2023/11), CORTEX (PID2021-123956OB-I00); funded by MCIN/AEI/10.13039/501100011033 and NL4DISMIS (CIPROM/2021/021).

References

Berk Atil, Sarp Aykent, Alexa Chittams, Lisheng Fu, Rebecca J. Passonneau, Evan Radcliffe, Guru Rajan Rajagopal, Adam Sloan, Tomasz Tudrej, Ferhan Ture, Zhe Wu, Lixinyu Xu, and Breck Baldwin. 2025. Non-determinism of "deterministic" llm settings.

Martin Atzmüller, Peter Klügl, and Frank Puppe. 2008. Rule-based information extraction for structured data acquisition using textmarker. In *LWA*.

Hira Lal Bhandari and Roshan Chitrakar. 2024. Enhancement of a transformation algorithm to migrate sql database into nosql graph database. *Data Science Journal*.

Jiaoyan Chen, Ernesto Jimenez-Ruiz, Ian Horrocks, and Charles Sutton. 2019. Learning semantic annotations for tabular data.

Zhikai Chen, Han Xie, Jian Zhang, Xiang song, Jiliang Tang, Huzefa Rangwala, and George Karypis. 2025. Autog: Towards automatic graph construction from tabular data.

Marco Cremaschi, Fabio D'Adda, and Andrea Maurino. 2025. steellm: An llm for generating semantic annotations of tabular data. *ACM Trans. Intell. Syst. Technol.* Just Accepted.

Marco Cremaschi, Blerina Spahiu, Matteo Palmonari, and Ernesto Jimenez-Ruiz. 2024. Survey on semantic interpretation of tabular data: Challenges and directions.

Roberto De Virgilio, Antonio Maccioni, and Riccardo Torlone. 2013. Converting relational to graph databases. In *First International Workshop on Graph Data Management Experiences and Systems*, GRADES '13, New York, NY, USA. Association for Computing Machinery.

Yuyang Dong, Chuan Xiao, Takuma Nozawa, Masafumi Enomoto, and Masafumi Oyamada. 2023. Deepjoin: Joinable table discovery with pre-trained language models.

Shengda Fan, Xin Cong, Yuepeng Fu, Zhong Zhang, Shuyan Zhang, Yuanwei Liu, Yesai Wu, Yankai Lin, Zhiyuan Liu, and Maosong Sun. 2024. Workflowllm: Enhancing workflow orchestration capability of large language models.

Chen Feng, Yifan Li, Zhaoda Chen, and Longxing Guo. 2025. The evolution and breakthrough of natural language processing: The revolution from rules to deep learning. In *Proceedings of the 2024 5th International Conference on Computer Science and Management Technology*, ICCSMT '24, page 307–311, New York, NY, USA. Association for Computing Machinery.

N. I. Galanis, P. Vafiadis, K. G. Mirzaev, and G. A. Papakostas. 2021. Machine learning meets natural language processing – the story so far.

Yury Gorishniy, Ivan Rubachev, and Artem Babenko. 2023. On embeddings for numerical features in tabular deep learning.

Sireesh Gururaja, Amanda Bertsch, Clara Na, David Widder, and Emma Strubell. 2023. To build our future, we must know our past: Contextualizing paradigm shifts in natural language processing. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, page 13310–13325. Association for Computational Linguistics.

Fiona Harrington, Elliot Rosenthal, and Miles Swinburne. 2024. Mitigating hallucinations in large language models with sliding generation and self-checks.

Zhengyong Jiang, Jionglong Su, Tong Chen, Zimu Wang, and Procheta Sen. 2024. Knowledge base-enhanced multilingual relation extraction with large language models. In *LKM2024: The First International OpenKG Workshop Large Knowledge-Enhanced Models @IJCAI 2024*.

Amin Karbasi, Omar Montasser, John Sous, and Grigoris Velegkas. 2025. (im)possibility of automated hallucination detection in large language models.

Safae Laqrichi. 2024. A hybrid framework for cosmic measurement: Combining large language models with a rule-based system.

Si-Yang Liu, Qile Zhou, and Han-Jia Ye. 2025. Make still further progress: Chain of thoughts for tabular data leaderboard.

Tianyang Liu, Fei Wang, and Muhao Chen. 2023. Rethinking tabular data understanding with large language models.

Scott Miller, David Stallard, Robert Bobrow, and Richard Schwartz. 1996. A fully statistical approach to natural language interfaces. In *34th Annual Meeting of the Association for Computational Linguistics*, pages 55–61, Santa Cruz, California, USA. Association for Computational Linguistics.

Jio Oh, Geon Heo, Seungjun Oh, Hyunjin Kim, JinYeong Bak, Jindong Wang, Xing Xie, and Steven Euijong Whang. 2025. Better think with tables: Tabular structures enhance llm comprehension for data-analytics requests.

I Made Putrama and Péter Martinek. 2022. An automated graph construction approach from relational databases to neo4j. In *2022 IEEE 22nd International Symposium on Computational Intelligence and Informatics and 8th IEEE International Conference on Recent Achievements in Mechatronics, Automation, Computer Science and Robotics (CINTI-MACRo)*, pages 000131–000136.

Yifan Song, Guoyin Wang, Sujian Li, and Bill Yuchen Lin. 2024. The good, the bad, and the greedy: Evaluation of llms should not ignore non-determinism.

Weihang Su, Changyue Wang, Qingyao Ai, Yiran Hu, Zhijing Wu, Yujia Zhou, and Yiqun Liu. 2024. Unsupervised real-time hallucination detection based on the internal states of large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 14379–14391, Bangkok, Thailand. Association for Computational Linguistics.

Yuan Sui, Mengyu Zhou, Mingjie Zhou, Shi Han, and Dongmei Zhang. 2024a. Table meets llm: Can large language models understand structured table data? a benchmark and empirical study.

Yuan Sui, Jiaru Zou, Mengyu Zhou, Xinyi He, Lun Du, Shi Han, and Dongmei Zhang. 2024b. Tap4llm: Table provider on sampling, augmenting, and packing semi-structured data for large language model reasoning.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, Louis Rouillard, Thomas Mesnard, Geoffrey Cideron, Jean bastien Grill, Sabela Ramos, Edouard Yvinec, Michelle Casbon, Etienne Pot, Ivo Penchev, Gaël Liu, Francesco Visin, Kathleen Kenealy, Lucas Beyer, Xiaohai Zhai, Anton Tsitsulin, Robert Busa-Fekete, Alex Feng, Noveen Sachdeva, Benjamin Coleman, Yi Gao, Basil Mustafa, Iain Barr, Emilio Parisotto, David Tian, Matan Eyal, Colin Cherry, Jan-Thorsten Peter, Danila Sinopalnikov, Surya Bhupatiraju, Rishabh Agarwal, Mehran Kazemi, Dan Malkin, Ravin Kumar, David Vilar, Idan Brusilovsky, Jiaming Luo, Andreas Steiner, Abe Friesen, Abhanshu Sharma, Abheesht Sharma, Adi Mayrav Gilady, Adrian Goedeckemeyer, Alaa Saade, Alex Feng, Alexander Kolesnikov, Alexei Bendebury, Alvin Abdagic, Amit Vadi, András György, André Susano Pinto, Anil Das, Ankur Bapna, Antoine Miech, Antoine Yang, Antonia Paterson, Ashish Shenoy, Ayan Chakrabarti, Bilal Piot, Bo Wu, Bobak Shahriari, Bryce Petrini, Charlie Chen, Charline Le Lan, Christopher A. Choquette-Choo, CJ Carey, Cormac Brick, Daniel Deutsch, Danielle Eisenbud, Dee Cattle, Derek Cheng, Dimitris Paparas, Divyashree Shivakumar Sreepathihalli, Doug Reid, Dustin Tran, Dustin Zelle, Eric Noland, Erwin Huizenga, Eugene Kharitonov, Frederick Liu, Gagik Amirkhanyan, Glenn Cameron, Hadi Hashemi, Hanna Klimczak-Plucińska, Harman Singh, Harsh Mehta, Harshal Tushar Lehri, Hussein Hazimeh, Ian Ballantyne, Idan Szpektor, Ivan Nardini, Jean Pouget-Abadie, Jetha Chan, Joe Stanton, John Wieting, Jonathan Lai, Jordi Orbay, Joseph Fernandez, Josh Newlan, Ju yeong Ji, Jyotinder Singh, Kat Black, Kathy Yu, Kevin Hui, Kiran Vodrahalli, Klaus Greff, Linhai Qiu, Marcella Valentine, Marina Coelho, Marvin Ritter, Matt Hoffman, Matthew Watson, Mayank Chaturvedi, Michael Moynihan, Min Ma, Nabila Babar, Natasha Noy, Nathan Byrd, Nick Roy, Nikola Momchev, Nilay Chauhan, Noveen Sachdeva, Oskar Bunyan, Pankil Botarda, Paul Caron, Paul Kishan Rubenstein, Phil Culliton, Philipp Schmid, Pier Giuseppe Sessa, Pingmei Xu, Piotr Stanczyk, Pouya Tafti, Rakesh Shivanna, Renjie Wu, Renke Pan, Reza Rokni, Rob Willoughby, Rohith Vallu, Ryan Mullins, Sammy Jerome, Sara Smoot, Sertan Girgin, Shariq Iqbal, Shashir Reddy, Shruti Sheth, Siim Põder, Sijal Bhatnagar, Sindhu Raghuram Panyam, Sivan Eiger, Susan Zhang, Tianqi Liu, Trevor Yacovone, Tyler Liechty, Uday Kalra, Utku Evci, Vedant Misra, Vincent Roseberry, Vlad Feinberg, Vlad Kolesnikov, Woohyun Han, Woosuk Kwon, Xi Chen, Yinlam Chow, Yuvein Zhu, Zichuan Wei, Zoltan Egyed, Victor Cotruta, Minh Giang, Phoebe Kirk, Anand Rao, Kat Black, Nabila Babar, Jessica Lo, Erica Moreira, Luiz Gustavo Martins, Omar Sanseviero, Lucas Gonzalez, Zach Gleicher, Tris Warkentin, Vahab Mirrokni, Evan Senter, Eli Collins, Joelle Barral, Zoubin Ghahramani, Raia Hadsell, Yossi Matias, D. Sculley, Slav Petrov, Noah Fiedel, Noam Shazeer, Oriol Vinyals, Jeff Dean, Demis Hassabis, Koray Kavukcuoglu, Clement Farabet, Elena Buchatskaya, Jean-Baptiste Alayrac, Rohan Anil, Dmitry, Lepikhin, Sebastian Borgeaud, Olivier Bachem, Armand Joulin, Alek Andreev, Cassidy Hardin, Robert Dadashi, and Léonard Hussenot. 2025. Gemma 3 technical report.

Mohamed Trabelsi, Zhiyu Chen, Shuo Zhang, Brian D. Davison, and Jeff Heflin. 2022. Strubert: Structure-aware bert for table search and matching. In *Proceedings of the ACM Web Conference 2022*, WWW '22, page 442–451, New York, NY, USA. Association for Computing Machinery.

Jialin Wang and Zhihua Duan. 2024. Agent ai with langgraph: A modular framework for enhancing machine translation using large language models.

Zilong Wang, Hao Zhang, Chun-Liang Li, Julian Martin Eisenschlos, Vincent Perot, Zifeng Wang, Lesly Miculicich, Yasuhisa Fujii, Jingbo Shang, Chen-Yu Lee, and Tomas Pfister. 2024. Chain-of-table: Evolving tables in the reasoning chain for table understanding.

Gerhard Weikum. 2002. Foundations of statistical natural language processing. *SIGMOD Rec.*, 31(3):37–38.

Xumeng Wen, Shun Zheng, Zhen Xu, Yiming Sun, and Jiang Bian. 2025. Scalable in-context learning on tabular data via retrieval-augmented large language models.

xAI. 2025. Grok 4. https://docs.x.ai/docs/models/grok-4-0709. Large language model by xAI.

Yunjia Zhang, Jordan Henkel, Avrilia Floratou, Joyce Cahoon, Shaleen Deep, and Jignesh M. Patel. 2023. Reactable: Enhancing react for table question answering.