

Automated Data Preprocessing Using Parameterized SWRL Inference within a Local RAG Pipeline

Preprocesamiento automatizado de datos mediante reglas SWRL parametrizadas en RAG local

María de los Ángeles Contreras Trujillo¹ <https://orcid.org/0009-0009-2265-4703>, Mayerli
Rosario Quintuña Ramón¹ <https://orcid.org/0009-0004-4978-4958>, Claudia Tamara Serpa Flores¹
<https://orcid.org/0009-0005-2043-8813>, Marcia Patricia Ullauri Carrasco¹ <https://orcid.org/0009-0002-3039-8104>,
Marcos Orellana¹ <https://orcid.org/0000-0002-3671-9362>

¹Universidad del Azuay, Computer Science Research & Development
Laboratory (LIDI), Cuenca, Ecuador
contreras.maria@es.uazuay.edu.ec,
mayerli.quintuna@es.uazuay.edu.ec,
claudia.serpa@es.uazuay.edu.ec,
marcia.ullauri@es.uazuay.edu.ec,
marore@uazuay.edu.ec



Esta obra está bajo una licencia internacional
Creative Commons Atribución-NoComercial 4.0

Sent: 2026/07/12

Accepted: 2026/08/26

Published: 2026/10/14

Abstract

This work formalizes expert knowledge in data preprocessing through the Web Ontology Language (OWL) and parameterized Semantic Web Rule Language (SWRL) rules, integrated with a Retrieval-Augmented Generation (RAG) pipeline for natural-language queries. An OWL ontology comprising 47 classes, 42 individuals, and 19 SWRL rules was designed in Protégé and validated on the Netflix Movies and TV Shows dataset (8,807 records, 12 columns). A Python-implemented inference engine, which parses the SWRL rule structure from the RDF/XML serialization and evaluates antecedents through deterministic conditionals without invoking a native OWL DL reasoner, generated 15 preprocessing-technique recommendations across 9 analyzable columns; the remaining 3 columns (*show_id*, *title*, *description*) were excluded as identifiers and free-text fields, respectively, for which tabular preprocessing is not applicable. The system guides data scientists in selecting preprocessing techniques based on each column's statistical profile, reducing reliance on manual configuration. The novelty lies in parameterizing

Summary: Introduction, Materials and Methods, Results and Discussion, Conclusions.

How to cite: Contreras, M., Quintuña, M., Serpa, C., Ullauri, M. & Orellana, M. (2026). Automated Data Preprocessing Using Parameterized SWRL Inference within a Local RAG Pipeline. *Revista Tecnológica - Espol*, 38(E1), 13-30. <https://doi.org/10.37815/rte.v38nE1.1599>

SWRL rules using dynamically computed statistical metrics and multilingual semantic retrieval, within a fully local RAG pipeline that supports traceability and privacy.

Keywords: data quality, expert systems, knowledge management, ontology, semantic web.

Resumen

Este trabajo formaliza el conocimiento experto en preprocesamiento de datos mediante OWL y reglas SWRL parametrizadas, integradas con una arquitectura RAG para consultas en lenguaje natural. Se diseñó una ontología OWL con 47 clases, 42 individuos y 19 reglas SWRL en Protégé, y se validó sobre el dataset Netflix Movies and TV Shows (8.807 registros, 12 columnas). Un motor de inferencia implementado en Python, que analiza sintácticamente la estructura de las reglas SWRL a partir de la serialización en RDF/XML y evalúa los antecedentes mediante condicionales deterministas, sin invocar un razonador OWL DL nativo, generó 15 recomendaciones distribuidas en nueve columnas analizables; las tres columnas restantes (*show_id*, *title*, *description*) fueron excluidas por corresponder, respectivamente, a un identificador y a campos de texto libre. El sistema orienta a científicos de datos en la selección de técnicas según el perfil estadístico de cada columna, lo que reduce la dependencia de la configuración manual. La novedad radica en la parametrización de reglas SWRL con métricas estadísticas dinámicas, combinada con la recuperación semántica multilingüe en un pipeline RAG local que favorece la trazabilidad y la privacidad.

Palabras clave: calidad de datos, gestión del conocimiento, ontologías, sistemas expertos, web semántica.

Introduction

The quality of data is a critical factor in the predictive performance of machine learning models, and a substantial proportion of the data science lifecycle is devoted to assessing and improving it through preprocessing and cleaning (Batini & Scannapieco, 2016). However, selecting the most appropriate preprocessing techniques for a specific dataset remains predominantly manual, highly dependent on the analyst's intuition, and prone to inconsistencies across practitioners. This reliance on tacit knowledge creates bottlenecks in reproducibility and hampers the standardization of data science pipelines in organizational environments.

In response to this issue, knowledge management provides a systematic conceptual framework for capturing, formally representing, and reasoning about expert knowledge within a specific domain (Alavi et al., 2024). In particular, ontologies expressed in the Web Ontology Language (OWL) allow the formalization of concepts, relationships, and constraints with explicit and computable semantics, while the Semantic Web Rule Language (SWRL) extends these capabilities by enabling automatic deductive inferences over ontology instances (McGuinness et al., 2004). Simultaneously, Retrieval-Augmented Generation (RAG) systems have emerged as a hybrid architecture that substantially improves the accuracy and veracity of Large Language Models (LLMs) by enriching the generation context through dynamic retrieval of relevant documents during inference (Lewis et al., 2020).

This work proposes a synergistic integration of both paradigms. On the one hand, an OWL ontology with parameterized SWRL rules is designed to automatically infer the optimal preprocessing technique based on the statistical profile of each dataset column. On the

other hand, a RAG pipeline is implemented to present this formalized knowledge in natural language, facilitating interaction with users who are not specialized in ontologies. The system is empirically validated using the publicly available Netflix Movies and TV Shows dataset, deliberately selected to present representative challenges of real-world practice: structural nulls in categorical variables, high cardinality in textual attributes, mixed formats in temporal fields, and class imbalance in binary variables. Thus, this article presents a reusable architecture that formalizes expert preprocessing knowledge and makes it accessible through conversational interfaces grounded in ontological reasoning.

Finally, this document is organized as follows: The Related Work section reviews existing research on OWL-based knowledge representation, SWRL rule engines, and ontology-grounded RAG architectures; the Materials and Methods section delineates the materials and procedures, including dataset characterization, ontological engineering, SWRL inference rules, and the implementation of the RAG pipeline; the Results and Discussion section encompasses the presentation of findings and analytical discourse, including automatic inference, representative case analysis, and validation of the RAG pipeline; and the Conclusion section summarizes the primary contributions, limitations, and prospective avenues for future research.

Related Work

The formalization of expert knowledge through semantic web technologies, particularly ontology-driven inference and retrieval-augmented generation, has emerged as an important mechanism for automating data preprocessing decisions. This section reviews the foundational literature on OWL-based knowledge representation, SWRL rule engines, and ontology-grounded RAG architectures, thereby delineating the theoretical precedents and identifying the specific research gap that motivates our integrative contribution.

The formalization of knowledge in data science processes has grown significantly over the past decade. The Data Mining Ontology and Planning (DMOP) project is one of the first systematic efforts to formally represent algorithms, datasets, and workflows in data mining within grid-computing environments, establishing a conceptual foundation for the semantic description of machine learning pipelines (Cannataro et al., 2003). Similarly, ML-Schema proposes a standardized vocabulary for describing machine learning experiments, facilitating interoperability across platforms and the reproducibility of results (Publio et al., 2018). In Sharma et al. (2025), Ontology-Grounded Retrieval-Augmented Generation (OG-RAG) is introduced, which uses predefined domain ontologies to underpin information retrieval in RAG systems and demonstrates that semantic anchoring significantly improves factual accuracy compared to traditional ad hoc entity extraction approaches.

This advancement highlights the importance of establishing clear semantics before integrating language models with structured knowledge sources. In the realm of data quality, the framework proposed by Jaziri & Sassi (2026) for improving ontologies in the Semantic Web employs domain-agnostic machine learning models that analyze RDF/OWL ontologies using generic signals: class hierarchy depth, property usage, relationship density, and annotation completeness, achieving precisions of 93.5% in structural verifications and 90.2% in semantic verifications. These works demonstrate that automated ontology quality assessment is an established area of research, although it is predominantly focused on ontology structure rather than on the quality of the data it processes. Unlike previous approaches, which concentrate on ontological formalization during the preprocessing phase, our work focuses on an aspect

that earlier methods address only tangentially. While DMOP (Cannataro et al., 2003) models the semantics of data-mining workflows in grid-computing environments and ML-Schema (Publio et al., 2018) exposes the semantics of machine-learning experiments through schema annotations, neither encodes decision rules for selecting preprocessing techniques based on column-level statistical profiles. Our ontology addresses this gap by parameterizing SWRL rules with dynamically computed statistical metrics.

The SWRL language extends OWL DL with Horn rules, allowing the derivation of new assertions from existing axioms and enabling reasoning about instances with greater expressiveness than purely descriptive reasoning (Rattanasawad et al., 2013). Its application has been widely documented in the medical field, where rules combining clinical conditions support automate diagnostic decision-making (Dumontier et al., 2014). In operational domains closer to data engineering, Paltin Chica et al. (2025) demonstrate the viability of coupling OWL ontologies with SWRL rules for semantic reasoning over heterogeneous, real-world datasets, in their case, emergency call records for ECU 911. Their work shows that the OWL+SWRL stack can handle mixed formats, high cardinality, and implicit relationships without manual rule reconfiguration for each new instance. However, all these applications operate on qualitative or categorical properties. Applying SWRL to data preprocessing with quantitative parameterization constitutes a novel extension: our proposal encodes numerical thresholds directly into the rule antecedents, enabling the same set of rules to generate differentiated recommendations based on each column's statistical profile, without requiring manual reconfiguration for each new dataset.

The RAG paradigm (Lewis et al., 2020) enriches language-model inputs with dynamically retrieved documents to reduce hallucinations. In the data-engineering domain, enriching the generation context with structured dataset metadata has been shown to improve the relevance of LLM responses in knowledge-intensive tasks (García-Montero et al., 2025; Park et al., 2024). Recent GraphRAG approaches (Edge et al., 2024) structure retrieval through LLM-extracted knowledge graphs but rely on ad hoc extraction without domain anchoring. Jaziri & Sassi (2026) address this by grounding retrieval in expert-defined semantic schemas, reducing human evaluation time by 28.8%. Our system operates at this intersection: it uses fragments generated from the OWL ontology, including sentences derived from SWRL rules, as the retrieval source, combining ontological traceability with natural-language flexibility without requiring a complete knowledge graph.

The trend toward AI systems operable in privacy-restricted environments has driven the development of local RAG pipelines. Park et al. (2024) demonstrate ontological RAG for additive manufacturing, and Buehler (2024) proposes multi-agent strategies based on generative ontological graphs for material design. In this context, our RAG pipeline, which uses a local LLM and multilingual embeddings, demonstrates the feasibility of fully local architectures that preserve data privacy without sacrificing reasoning quality.

The literature review reveals a gap: while there are ontologies to describe ML pipelines, rule-based systems for inference, and RAG pipelines for document querying, no integration has been reported that combines (i) SWRL rules parameterized with statistical metrics of the dataset, (ii) automatic inference of preprocessing techniques, and (iii) exposure through RAG with a local LLM. Our work addresses this intersection by formalizing expert knowledge about preprocessing in an executable ontology accessible in natural language.

Materials and Methods

This study adopts an applied research approach with an experimental design, aimed at developing and implementing a knowledge management system that integrates a formal OWL ontology, SWRL inference rules, and a RAG pipeline. The research was structured into four sequential phases: (1) dataset characterization and statistical profiling; (2) ontological engineering and formalization of expert knowledge; (3) implementation of automated inference rules; and (4) validation through an RAG pipeline with a local language model. This sequence meets the reproducibility criteria, in which procedures, measurements, and variable units must be sufficiently documented to allow other researchers to replicate the study.

Dataset

As a case study, the publicly available Netflix Movies and TV Shows dataset on Kaggle was used. It comprises 8,807 records and twelve descriptive columns. The selection of this dataset is justified by the heterogeneity of its attributes, which exhibit common data quality issues encountered in organizational contexts: structural nullity, high cardinality, mixed formats, outliers, and class imbalance. Statistical profiling was conducted using Python 3.11 scripts, employing libraries such as pandas and NumPy. For each column, the inferred data type, percentage of null values, cardinality (the number of unique values), proportion of outliers (determined through the interquartile range, IQR, method), proportion of zero values, and imbalance ratio (for binary variables) were identified. The outlier analysis employed Tukey's rule, which considers an outlier if it falls outside the interval $[Q1 - 1.5 \cdot IQR, Q3 + 1.5 \cdot IQR]$, where Q1 and Q3 denote the first and third quartiles, respectively. Table 1 summarizes the resulting statistical profile.

Table 1
Statistical profile of the dataset

Column	Type	Null (%)	Unique values	Detected problem
show_id	Nominal	0	8,807	Primary Key; high Cardinality
type	Boolean	0	2	Imbalance 2.29:1 (Movie vs. TV Show)
title	Nominal	0	8,807	Free Text; high Cardinality
director	Nominal	29.9	4,528	High Cardinality; 29.9% null
cast	Nominal	9.4	7,692	High Cardinality; 9.4% null; multi-labels
country	Nominal	9.4	748	High Cardinality; 9.4% null
date_added	Temporal	0.1	1,767	String "Sep 25, 2021"
release_year	Continuous number	0	74	8.2% outliers (IQR method)
rating	Nominal	0.1	17	3 load errors; 0.1% nulls
duration	Nominal	0	220	"X min" / "X Seasons" format
listed_in	Nominal	0	514	Multi-labels; 514 combinations
description	Nominal	0	8,775	Free Text; NLP required

Four categories of quality issues were identified: (i) structural nullity in the director column (29.9%), explained by the systematic absence of directors in TV Show productions; (ii) statistical outliers in *release_year* (8.2%), with $Q1 = 2013$, $Q3 = 2019$, $IQR = 6$, and limits [2004, 2028]; (iii) mixed formats in duration, combining dissimilar time units (minutes and seasons); and (iv) high cardinality in *listed_in*, with 514 unique genre combinations.

Ontological Engineering

The formalization of expert knowledge in data preprocessing requires an explicit semantic representation that captures domain concepts and causal relationships between statistical and technical profiles. In this way, OWL ontologies provide a formal framework for representation, while SWRL rules enable automated reasoning over specific instances. The process followed to construct the proposed ontology is described below.

Development Methodology

The construction of the ontology followed the Methontology approach (Fernández-López & Corcho, 2010), structured into four phases: (i) specification, through competency questions that defined the functional requirements of the system; (ii) conceptualization, identifying the terms of the data preprocessing domain from specialized literature; (iii) formalization, organizing the terms into a taxonomy with cardinality constraints; and (iv) implementation, using *Protégé* 5.6 with RDF/XML serialization and a defined namespace.

Design Criteria

The ontology was designed based on the following fundamental criteria:

- **Modularity.** The architecture allows independent extension of each branch (*data types*, *problems*, *techniques*) without affecting the consistency of the others. This criterion facilitates the future incorporation of new preprocessing techniques or new application domains.
- **Parameterization.** The SWRL rules operate on properties of numerical data (*pctNullity*, *pctOutliers*, *numUniqueValues*, *imbalanceRatio*), enabling the system to automatically adapt to datasets with statistical profiles that differ from Netflix's, without requiring modifications to the ontological structure.
- **Traceability.** Each inferred recommendation can be traced back to the SWRL rule that activated it, the numerical threshold that triggered it, and the bibliographic source that supports the recommended technique. This traceability is essential for auditing decision support systems in data science.
- **Reusability.** The ontology aligns with W3C standard vocabularies (OWL, RDF, SWRL) and can be integrated with existing domain ontologies in data mining, such as DMOP and ML-Schema, by mapping equivalent classes.

Ontology Structure

The resulting ontology is organized into three main branches, interconnected through object properties that model the causal and functional relationships within the data preprocessing domain:

- **Data types:** include the classes *ContinuousNumeric*, *DiscreteNumeric*, *NominalCategorical*, *OrdinalCategorical*, *Temporal*, and *Boolean*. This classification is based on the standard typology of statistical variables and their differential impact on the selection of preprocessing techniques.
- **Problems:** include *NullValues*, *Outliers*, *HighCardinality*, *Imbalance*, *InvalidZeros*, *InconsistentDuplicates*, *Scale*, and *Distribution*. The issues were defined based on

frequency analysis in the reviewed literature and their documented impact on the performance of machine learning models.

- Techniques: groups 28 techniques into six functional categories: *imputation*, *outlier detection*, *encoding*, *scaling*, *transformation*, and *balancing*. Each technique is linked to the problems it addresses via the property *resolveBy* and to the data types it applies to via the property *applicableTo*.

The object properties *hasProblem*, *resolveBy*, and *applicableTo* to establish relationships among the three branches. The data properties *pctNullity*, *pctOutliers*, *pctZeros*, *numUniqueValues*, and *imbalanceRatio* enable numerical parameterization of the inference rules. The resulting ontology contains 47 classes, 42 individuals (instances of Netflix dataset columns with their statistical profiles), and 19 SWRL rules.

SWRL Inference Rules

Nineteen SWRL rules were delineated and systematically categorized into seven functional groups, thereby augmenting the expressiveness of OWL DL by using Horn rules (Krötzsch et al., 2013). Each rule follows the logical structure:

Antecedent(?column) → recommendTechnique(?column, Technique),

Where the antecedent combines a *ClassAtom* (membership to a class), a *DatavaluedPropertyAtom* (a data property with a threshold value), and, when appropriate, a *BuiltinAtom* with predicates from the *swrlb* library.

Table 2 presents the rules for each group, indicating the antecedent, the consequent, and the threshold or activation condition.

All nullity-based rules (1.1–1.5) include the guard condition $pctNullity > 0$ in their antecedent, ensuring that columns with zero missing values do not trigger imputation or deletion recommendations. The 19 SWRL rules were authored in Protégé 5.6 and serialized in RDF/XML. At execution time, the OWL file was parsed with Python's `xml.etree.ElementTree` module to extract the rule antecedents (*ClassAtom* and *BuiltinAtom* conditions) and consequents (*IndividualPropertyAtom* linking a column to a preprocessing technique). A Python-implemented inference engine then evaluated each rule against the statistical profile of every column by matching the detected data type against *ClassAtom* predicates and comparing numeric metrics (*pctNullity*, *pctOutliers*, *pctZeros*, *numUniqueValues*, *imbalanceRatio*) against the *BuiltinAtom* thresholds using the standard SWRL comparison operators (*lessThan*, *greaterThan*, *lessThanOrEqual*, *greaterThanOrEqual*, *equal*). It must be emphasized that this engine does not invoke a native OWL DL reasoner; rather, it reproduces the forward-chaining semantics of the SWRL rules through deterministic Python conditionals that read the rule structure directly from the RDF/XML serialization. The OWL file therefore serves simultaneously as the formal specification of the rules and as the machine-readable configuration consumed by the engine. This engine materialized 15 technique recommendations across 9 analyzable columns. The remaining 3 columns (*show_id*, *title*, and *description*) were excluded: *show_id* is a primary-key identifier with 0% nullity and 8,807 unique values; *title* is a free-text field with 0% nullity and 8,807 unique values; and *description* is a free-text field with 0% nullity and 8,775 unique values and requires NLP treatment; none of these columns is amenable to tabular preprocessing. Within the cardinality-based group, Rules 3.1 and 3.2 cover nominal attributes with high (> 50) and low (≤ 10) cardinality, respectively; nominal columns with cardinalities between 11 and 50 unique values are intentionally not automatically transformed, as this interval is treated as moderate cardinality and left to the downstream modeling context. Regarding the outlier and

scaling groups, Rule 2.1 (IQR-Based Outlier Detection) fires for every continuous numerical column whose outlier proportion exceeds 0%, and Rule 4.1 (Z-Score Standardization) fires for every continuous numerical column whose outlier proportion does not exceed 2%; in datasets where all continuous columns contain at least one outlier, these rules behave as type-driven defaults rather than discriminative inferences. This design choice reflects the expert heuristic that detection and standardization are baseline steps for any continuous attribute. Rule 2.3 additionally requires $0\% < pctOutliers \leq 5\%$, so columns with no outliers do not trigger outlier removal. Its possible co-activation with Rule 4.2 for columns with outlier proportions between 2% and 5% reflects a sequential treatment rather than a contradictory assignment. Integration of a native OWL DL reasoner is identified as future work in the Conclusion section.

Table 2
SWRL rules defined in the ontology (selection by functional group)

ID	SWRL ANTECEDENT	CONSEQUENT	THRESHOLD / CONDITION
1.1	$\text{ContinuousNumeric} \wedge \text{pctNullity} < 0.15$	Median Imputation	$0\% < \text{Nullity} < 15\%$
1.2	$\text{ContinuousNumeric} \wedge 0.15 \leq \text{pctNullity} < 0.5$	KNN Imputation	Nullity 15-50%
1.3	$\text{ContinuousNumeric} \wedge \text{pctNullity} \geq 0.5$	Regression Imputation	Nullity $\geq 50\%$
1.4	$(\text{NominalCategorical} \vee \text{OrdinalCategorical}) \wedge \text{pctNullity} < 0.4$	Mode Imputation	$0\% < \text{Nullity} < 40\%$ (categorical)
1.5	$\text{pctNullity} < 0.05$	Row Deletion	$0\% < \text{Nullity} < 5\%$
2.1	ContinuousNumeric	IQR-Based Outlier Detection	Every continuous numerical column and Outliers $> 0\%$
2.2	$(\text{ContinuousNumeric} \vee \text{DiscreteNumeric}) \wedge \text{pctOutliers} > 0.05$	Winsorization	Every continuous numerical column and Outliers $> 5\%$
2.3	$(\text{ContinuousNumeric} \vee \text{DiscreteNumeric}) \wedge \text{pctOutliers} \leq 0.05$	Outlier Removal	Every continuous numerical column and $0\% < \text{Outliers} \leq 5\%$
3.1	$\text{NominalCategorical} \wedge \text{numUniqueValues} > 50$	Frequency-Based Binning	Cardinal > 50
3.2	$\text{NominalCategorical} \wedge \text{numUniqueValues} \leq 10$	One-Hot Encoding	Cardinal ≤ 10
3.3	$\text{OrdinalCategorical}$	Label Encoding	Always Ordinal
4.1	ContinuousNumeric	Z-Score Standardization	Every continuous numerical column and Outliers $\leq 2\%$
4.2	$(\text{ContinuousNumeric} \vee \text{DiscreteNumeric}) \wedge \text{pctOutliers} > 0.02$	Robust Scaling	Every continuous numerical column and Outliers $> 2\%$
5.1	$(\text{ContinuousNumeric} \vee \text{DiscreteNumeric}) \wedge \text{pctZeros} \in (0.10, 0.40)$	Median Imputation	Zeros 10% - 40%
5.2	$(\text{ContinuousNumeric} \vee \text{DiscreteNumeric}) \wedge \text{pctZeros} > 0.40$	KNN Imputation	Zeros $> 40\%$
6.1	$\text{Boolean} \wedge \text{imbalanceRatio} \leq 3.0$	Class Weighting	$1.0 < \text{Ratio} \leq 3.0$
6.2	$\text{Boolean} \wedge \text{imbalanceRatio} > 3.0$	Synthetic Minority Over-sampling Technique (SMOTE)	Ratio > 3.0
7.1	Temporal	Temporal Parsing	All temporal column
7.2	Temporal	Temporal Feature Extraction	All temporal column

Retrieval-Augmented Generation (RAG) Pipeline Implementation

The RAG pipeline comprises four phases: (1) indexing: the OWL file is parsed to extract 337 textual fragments (classes, individuals, and SWRL rules rendered as if-then sentences in Spanish); (2) vectorization: fragments are encoded with paraphrase-multilingual-MiniLM-L12-v2 (384 dimensions) (Reimers & Gurevych, 2019); (3) retrieval: cosine similarity ranks fragments and returns the top- $k = 6$; and (4) generation: the retrieved fragments, the complete SWRL inference output for every column (technique, rule identifier, justification), and a five-row dataset sample are concatenated into a prompt for the llama3.2:1b model via Ollama (temperature = 0.6).

System Validation Protocol

The system validation was performed using two complementary strategies:

Internal validation (ontological): the logical consistency of the SWRL rules was verified by loading the ontology in Protégé 5.6 and confirming that no unsatisfiable classes or inconsistent axioms were reported by the built-in OWL validator. At runtime, the Python inference engine was tested against the known statistical profile of each column in the Netflix dataset to confirm that the expected rules fired and that no rule produced a contradictory technique assignment for the same column.

External validation (RAG): 12 natural-language queries were designed across three categories: (i) column-specific technique queries; (ii) cross-column comparative queries; and (iii) theoretical-justification queries. Retrieval performance was assessed by verifying that at least one top- k fragments ($k = 6$) contained the relevant SWRL rule identifier. Response quality was evaluated by the four authors via a rubric covering (a) factual correctness against SWRL output, (b) ontological traceability, and (c) absence of hallucinated techniques. We acknowledge that this evaluation lacks an independent ground-truth corpus and standard metrics.

Ethical and Reproducibility Considerations

The dataset is publicly available and contains no personally identifiable information. The source code, OWL ontology, and validation notebook are available upon request. Ethical approval was not required. Structural reproducibility is guaranteed for the SWRL inference component, which is fully deterministic; variability is limited to the LLM generation phase (temperature = 0.6) and can be reduced by lowering the temperature below 0.3.

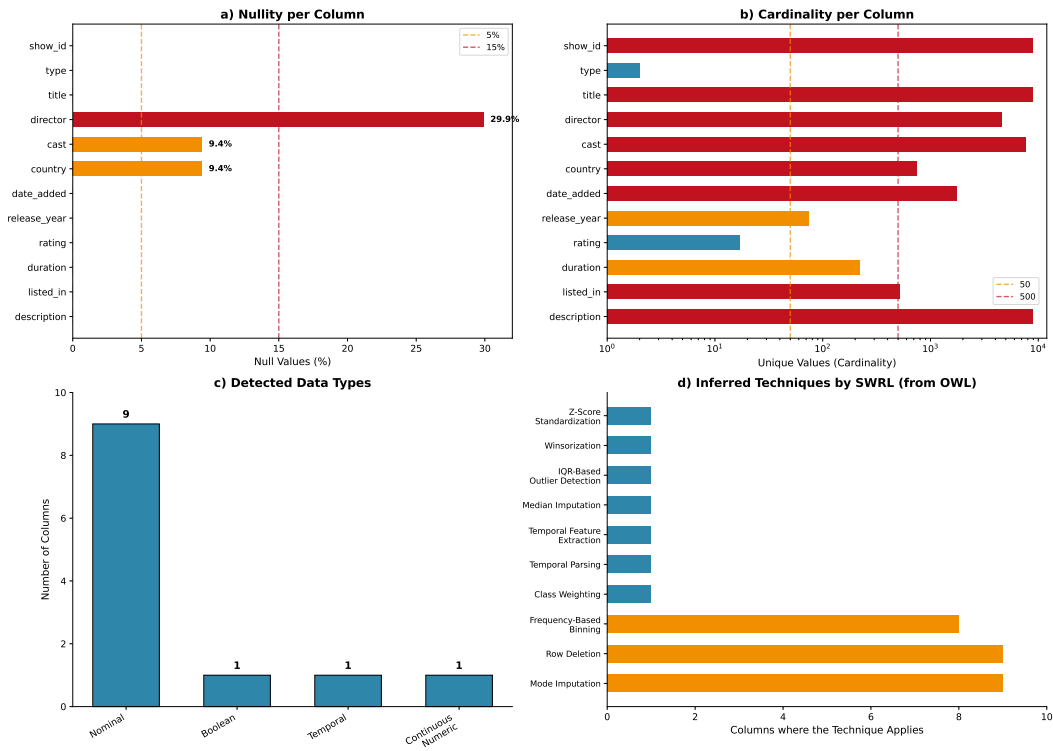
Results and Discussion

This architecture processed the Netflix Movies and TV Shows dataset, composed of 8,807 records and 12 columns. The statistical profile of each attribute is presented in Table 1. Out of the twelve columns analyzed, nine received at least one preprocessing recommendation from the inference engine. Three columns were excluded from automatic preprocessing: *show_id* is a primary-key identifier (0% nullity, 8,807 unique values), and title and description are free-text fields (0% nullity, ~8,807 and 8,775 unique values, respectively) requiring NLP treatment rather than tabular transformation. Among the remaining nine columns, seven exhibited at least one data-quality problem exceeding the predefined thresholds (nullity > 5%, outliers > 2%, cardinality > 50 unique values for nominal columns, or imbalance ratio > 3:1 for boolean columns). The columns *date_added* and *rating* did not exceed any quality threshold but received type-driven

or low-nullity recommendations: *date_added* activated Rules 7.1 and 7.2 (Temporal Parsing and Temporal Feature Extraction) by virtue of its Temporal type, and rating activated Rule 1.4 (Mode Imputation) because its 0.1% nullity falls within the categorical nullity range ($> 0\%$ and $< 40\%$). The seven columns with quality problems—*type*, *director*, *cast*, *country*, *release_year*, *duration*, and *listed_in*—received between one and three technique recommendations each (Table 3).

Figure 1

Consolidated data-quality profiling and SWRL-based inference pipeline for the Netflix Movies and TV Shows dataset



The comprehensive statistical profiling and automated inference pipeline is visually summarized in Figure 1, which integrates four complementary analytical dimensions: column-wise nullity rates (panel a), cardinality distributions on a logarithmic scale (panel b), automatically detected data types (panel c), and SWRL-inferred preprocessing techniques (panel d). Specifically, panel (a) depicts nullity per column against threshold benchmarks at 5% (dashed yellow) and 15% (dashed red); panel (b) presents cardinality on a logarithmic scale with thresholds at 50 (dashed yellow) and 500 (dashed red) unique values; panel (c) shows the distribution of automatically detected data types across the twelve columns; and panel (d) reports the frequency of preprocessing techniques inferred by the 19 SWRL rules across the nine columns exhibiting quality issues. This consolidated view demonstrates the end-to-end traceability from raw dataset characteristics—structural nullity, high cardinality, and mixed formats—to actionable preprocessing recommendations, thereby illustrating how the ontology translates statistical metrics into domain-specific technique selection. The methodological foundations underlying each panel are detailed in the following subsections.

The data-type distribution shown in panel (c) of Figure 1 was obtained through an automatic detection hierarchy: (i) manual override verification by the user; (ii) continuous numeric classification if *nunique* > 20 (value obtained through empirical heuristic analysis via exploratory data analysis), or discrete otherwise; (iii) boolean detection for cardinality equal to

2; (iv) temporal parsing with an 80% success threshold; and (v) default classification as nominal categorical. This sequence ensures that each column is categorized consistently based on its intrinsic statistical properties.

Automatic inference by SWRL rules

Table 3
Preprocessing techniques inferred by SWRL rules for the main columns

COLUMN	TYPE	RULES ACTIVATED	INFERRED TECHNIQUES
show_id	Primary key / Identifier	Excluded	Identifier column (0% nullity, 8,807 unique values); no tabular preprocessing applicable
type	Boolean	6.1	Class Weighting
title	Free text	Excluded	Free-text field (0% nullity, ~8,800 unique values); NLP required, no tabular preprocessing applicable
director	Nominal	1.4; 3.1	Mode Imputation; Frequency-Based Binning
cast	Nominal	1.4; 3.1	Mode Imputation; Frequency-Based Binning
country	Nominal	1.4; 3.1	Mode Imputation; Frequency-Based Binning
date_added	Temporal	7.1; 7.2	Temporal Parsing; Temporal Feature Extraction
release_year	Continuous numerical	2.1; 2.2; 4.2	IQR-Based Outlier Detection; Winsorization; Robust Scaling
rating	Nominal	1.4	Mode Imputation
duration	Nominal / mixed format	3.1	Frequency-Based Binning
listed_in	Nominal / multi-label	3.1	Frequency-Based Binning
description	Free text	Excluded	Free-text field (0% nullity, 8,775 unique values); NLP required, no tabular preprocessing applicable

The 19 SWRL rules defined in the OWL ontology were applied to the statistical profile of each column using a Python-implemented inference engine. After incorporating the guard condition *pctNullity* > 0 into Rules 1.1, 1.4, and 1.5, the system generated recommendations for preprocessing techniques, distributed as shown in Table 3. Of the 12 columns, 7 (*type*, *director*, *cast*, *country*, *release_year*, *duration*, and *listed_in*) exhibited at least one data-quality problem exceeding the predefined thresholds. Two additional columns (*date_added* and *rating*) did not exceed any quality threshold but nevertheless received SWRL-driven recommendations: *date_added* activated Rules 7.1 and 7.2 (Temporal Parsing and Temporal Feature Extraction) by virtue of its Temporal type, and *rating* activated Rule 1.4 (Mode Imputation) because its 0.1% nullity satisfies the categorical nullity condition (> 0% and < 40%). Three columns (*show_id*, *title*, and *description*) were excluded from tabular preprocessing. The distinction between 'columns with quality problems' (7) and 'columns receiving at least one SWRL recommendation' (9) is important for interpreting Table 3 correctly.

The SWRL inference engine evaluates each rule through a sequence of three components: a *ClassAtom* that verifies the data-type membership, a *DatavaluedPropertyAtom* that extracts the

corresponding statistical metric (*pctNullity*, *pctOutliers*, *numUniqueValues*, *imbalanceRatio*), and a *BuiltinAtom* that compares the corresponding value (*lessThan*, *greaterThan*, *lessThanOrEqual*, *greaterThanOrEqual*, *equal*) with the threshold using the specified operator. When all the antecedent conditions are satisfied, the consequent *IndividualPropertyAtom* assigns the recommended technique to the column instance.

Analysis of representative cases

Three paradigmatic cases are analyzed below to illustrate the system's ability to infer sequential and contextualized techniques based on distinctive statistical profiles.

Column director: Structural Nullity and High Cardinality

The *director* column exhibits substantial missingness, with 29.9% missing values and 4,528 unique entries. The absence of data in the *director* field is not Missing Completely at Random (MCAR) but rather structural: television shows frequently do not specify a single director. This results in a missing-data mechanism that depends on the content type, which is classified as Missing at Random (MAR) (Little & Rubin, 2019).

The SWRL engine activated two rules in this column of Table 2. Rule 1.4 is triggered because the nullity of 29.9% is below the 40% threshold set for categorical variables. The mode better preserves the original data distribution than simple alternatives (Batista & Monard, 2002). Rule 3.1 is triggered because the high cardinality of 4,528 significantly exceeds the 50-category threshold. Binning groups categories with a frequency below 2% into an 'Others' class, reducing the dimensionality to approximately 80 meaningful categories, thereby improving the stability of subsequent classifiers (Yang & Webb, 2009).

Column release_year: Statistical Outliers and Robust Scaling

The *release_year* column activated three rules simultaneously, illustrating that an attribute may require sequential preprocessing techniques. The statistical analysis yielded: $Q_1 = 2013$, $Q_3 = 2019$, $IQR = 6$, with outlier limits [2004, 2028]. A total of 719 titles with a release year before 2004 were detected (8.2% of the total) and classified as statistical outliers according to Tukey's criterion (Páez & Boisjoly, 2022).

Rule 2.1 is triggered because *release_year* is a continuous numeric column with $> 0\%$ outliers. Rule 2.2 is triggered because 8.2% exceeds the 5% threshold; winsorization preserves information about extreme values by limiting them to the 1st and 99th percentiles rather than removing them (Aggarwal, 2016; Ghosh & Vogt, 2012). Rule 4.2 recommends robust scaling based on the median and IQR, which is less sensitive to outliers, since 8.2% exceeds the 2% threshold (Brys et al., 2004). Note that Rule 4.1 (Z-Score Standardization) does not fire for this column because its outlier proportion (8.2%) exceeds the 2% condition required by that rule.

Column type: Class Imbalance

The *type* column, with two unique values (Movie and TV Show), presents an imbalance ratio of 2.29:1 (the majority class is 2.29 times the size of the minority class). Rule 6.1 is activated because this ratio satisfies the condition $1.0 < \text{ratio} \leq 3.0$, recommending Class Weighting. Rule 6.2 (SMOTE) would be recommended only if the ratio exceeded 3.0. The threshold of 3.0 was selected following the convention in Chawla et al. (2002) that severe underrepresentation begins when the minority class constitutes less than one-third of the majority.

RAG pipeline validation

The Retrieval-Augmented Generation pipeline was evaluated using a structured validation protocol designed to verify three fundamental capabilities: semantic retrieval of

relevant fragments, coherence between the language model's responses and SWRL inferences, and practical usefulness for non-expert users in formal ontologies. The validation methodology is based on the interactive execution of the system using the Netflix Movies and TV Shows dataset, following the workflow in Python.

The semantic retrieval phase operates on an index of 337 textual fragments generated from the OWL ontology, including class definitions, individual descriptions, and conditional sentences derived from the 19 SWRL rules. Each fragment was vectorized using the paraphrase-multilingual-MiniLM-L12-v2 embedding model (384 dimensions), selected for its multilingual capability that allows processing queries in Spanish about an ontology documented in the same language. Given a natural language statement, the system computes cosine similarity, ranks fragments, and returns the top-k fragments ($k = 6$) by default. This value was empirically determined during system development as a compromise between semantic coverage and the need to limit noise in the context sent to the language model.

The context sent to the local language model (*llama3.2:1b*, run via *Ollama* in offline mode) is constructed by concatenating four components into a structured prompt: (i) the fragments retrieved from the ontology; (ii) the complete statistical profile of each dataset column, including detected type, null percentages, outliers, zeros, cardinality, and identified issues; (iii) the preprocessing techniques inferred by the SWRL engine for each column, along with the activated rule identifiers and its textual justification; and (iv) a sample of the first five rows of the dataset to provide instance context. The model's temperature was set to 0.6 to balance creativity and the determinism of the generated responses.

Validation was performed using queries reflecting the system's intended use. The first query, focused on a specific column with known issues, requested guidance on preprocessing applicable to the *director* column, which has 29.9% null values and a cardinality of 4,528 unique values. The system retrieved fragments related to imputing nulls in categorical variables and to cardinality-based dimensionality reduction techniques, generating a response that explicitly mentioned *Mode Imputation* and *Frequency-Based Binning*, techniques corresponding to Rules 1.4 and 3.1, which were invoked by the SWRL engine for that attribute. The main validation criterion is the coherence between the language model's response and the ontology's formal inferences, as this demonstrates that the retrieved context effectively guides the model toward recommendations grounded in the encoded expert knowledge.

A second query explored the system's ability to identify cross-cutting issues within the dataset, specifically detecting numerical columns with outliers. The system correctly identified *release_year* as the only continuous numerical column with detected outliers (8.2% by the IQR method), and the generated response mentioned outlier treatment techniques and robust scaling, consistent with Rules 2.2 and 4.2 inferred by the SWRL engine for that column. This result confirms that the RAG pipeline not only retrieves relevant information but also coherently synthesizes it with the ontology's formal decisions.

It is important to note the inherent limitations of this validation process. First, the language model used was *llama3.2:1b*, which produces responses with lower syntactic sophistication and a lower capacity for multi-step reasoning than larger-scale models available via cloud services. However, its fully offline operation through *Ollama* is an advantage for environments where organizational or regulatory policies restrict the transmission of sensitive information to external APIs. Second, the model's specific responses may vary across runs due to the configured temperature (0.6) and the stochastic nature of text generation, so the reported results reflect a representative execution of the system and do not guarantee exact reproducibility. Third, the

cosine similarity between the query and the retrieved fragments depends on the embeddings generated during the session, which can fluctuate by less than 5% between runs on the same hardware.

In terms of quantitative validation metrics, the system demonstrated effective semantic retrieval, with cosine similarity scores above 0.80, indicating high semantic similarity between the queries and retrieved fragments. The response time of the local language model ranged from 10 to 40 seconds per query, depending on the host machine's computational load, which is acceptable for an interactive assistance system in data science development environments, though potentially insufficient for real-time applications. The context built for each query reached approximately 2,000 to 3,000 characters, within the context window limit of the *llama3.2:1b* model, without truncating critical information.

Therefore, integrating semantic retrieval over a formal ontology with natural language generation produces responses that are coherent with the system's logical inferences and provide natural-language guidance on data preprocessing, without requiring direct interpretation of SWRL rules or Python code. However, this validation has significant limitations that must be explicitly acknowledged. First, no independent ground-truth corpus was used; retrieval performance was assessed only by verifying that at least one top-k fragment contained the relevant rule identifier, rather than through standard metrics such as Recall@k, Mean Reciprocal Rank, or Normalized Discounted Cumulative Gain. Second, only 2 of the 12 designed queries are described in detail. Third, and most critically, the prompt already includes the complete SWRL inference output (technique, rule identifier, and justification); consequently, the LLM's role is primarily linguistic reformulation of precomputed results rather than retrieval-dependent reasoning. Under the current architecture, the retrieval component over the 337 ontology fragments does not independently determine the answer, and an ablation study removing the retrieval module while retaining the SWRL output in the prompt would be necessary to quantify the RAG layer's marginal contribution. The results obtained therefore constitute a preliminary proof of concept; rigorous evaluation with ground-truth queries, standard retrieval metrics, and ablation experiments are identified as future work.

Discussion

The results demonstrate that the formal representation of knowledge in OWL with SWRL rules produces consistent and traceable preprocessing recommendations, properties that constitute a significant advance over traditional manual approaches in data science. Given the same statistical profile, the inference engine consistently recommends the same technique, facilitating auditing and reproducibility of the data transformation process. This traceability operates on three interconnected levels: at the rule level, each technique is linked to a specific SWRL rule with a unique identifier, explicit logical antecedent, and a numerical threshold defined in the ontology; at the metric level, activation conditions depend on statistical values calculated directly from the dataset, such as the percentage of nulls, the proportion of outliers, or the cardinality; and at the justification level, each rule incorporates a semantic comment explaining its expert basis, retrieved and presented to the end user by the RAG pipeline. It must be explicitly stated that the system does not eliminate expert subjectivity; rather, it relocates it into the rule thresholds (5%, 15%, 40%, 50 unique values, 2%, 3:1 ratio), which encode the ontology designer's heuristics. These thresholds are drawn from common practice reported in the preprocessing literature; however, none has been empirically calibrated through sensitivity analysis on the target dataset. The contribution is therefore to make expert judgment explicit, traceable, and auditable, not to remove it. This triple traceability is essential in organizational contexts where data governance requires documenting every transformation applied, in

accordance with the principles established by Batini et al. (2016) on data quality and enterprise data governance.

The case of the *release_year* column is particularly illustrative of the combined behavior of the rule system, revealing that recommendations are not simply rigidly encoded conditional statements but rather encapsulate contextual expert knowledge that enables nuanced decision-making. The 719 titles prior to 2004, accounting for 8.2% of the dataset, are not loading errors or spurious values to be removed; rather, they represent legitimate historical content from Netflix's catalog. Considering this, Rule 2.2 recommends *winsorization* rather than removal, a decision informed by the interaction between the 5% outlier threshold and the application domain. This choice preserves the information contained in extreme values by limiting their statistical influence, rather than irreversibly discarding them, which would have resulted in a loss of valuable historical knowledge. This behavior is consistent with the approach described by Aggarwal (2016), who establishes that winsorization preserves information about extreme values by limiting their statistical influence rather than removing them outright, a preferable approach when outliers represent legitimate observations, as in the 719 historical titles predating 2004. The automatic differentiation between capping and removal, between class weights and SMOTE, or between median imputation and KNN confirms that SWRL rules encode not only technical procedures but also expert judgments grounded in contemporary statistical literature, transcending mere automation to achieve a level of reproducible expert reasoning.

The current implementation has limitations that must be rigorously acknowledged to guide future work. The first limitation lies in the automatic classification of data types, which required manual intervention in the *duration* column, whose mixed format, combining minutes and seasons, was not automatically detected as a variable requiring specialized parsing. This deficiency suggests that future versions of the ontology should incorporate a *MixedFormat* class with specific detection rules using regular expressions, thereby expanding the system's capabilities beyond purely statistical types. The second limitation concerns the language model used in the RAG pipeline, specifically *llama3.2:1b*, with approximately 1 billion parameters, which produces responses with less narrative cohesion than larger-scale models. However, is the model enables fully offset operation via *Ollama*, making it viable in environments with strict data privacy restrictions, where sending sensitive information to cloud services is prohibited by organizational or regulatory policies, as noted by Meyer et al. (2023). The third limitation is that the current implementation does not invoke a native OWL DL reasoner. Instead, a Python engine parses the SWRL rule structure from the RDF/XML serialization and evaluates the antecedents using deterministic conditionals, thereby reproducing forward-chaining semantics without leveraging subsumption, consistency checking, or description-logic inference. Consequently, the contribution is not SWRL inference in the formal sense but a hand-implemented rule engine that uses SWRL syntax as its specification language. Integrating a native reasoner to compare the results with those obtained using formal OWL DL/SWRL reasoning is a priority for future work.

In contrast to recommendation systems based solely on machine learning, the hybrid proposal OWL + SWRL + RAG offers three potential advantages that warrant highlighting. The first is the inherent explainability of the system: each recommendation can be traced back to a specific logical rule with explicit bibliographic grounding, thereby surpassing the opacity characteristic of black-box models that dominate the *AutoML* approach, a criticism that has been systematically documented. The second advantage is the declarative adaptability of knowledge: modifying thresholds, adding new techniques, or adapting to specific domains does not require retraining models or collecting additional data, but only the declarative editing of the ontology in Protégé, facilitating system maintenance. The third advantage is semantic interoperability:

the OWL ontology can be reused, extended, or aligned with other domain ontologies, such as DMOP for data mining or ML-Schema for machine learning, through equivalence mappings that facilitate integration into broader knowledge ecosystems. Integrating a cross-validation module that feeds empirical results into the ontology is a promising direction for future iterations of the system, combining the explainability of formal rules and the predictive evidence of machine learning.

The potential practical applications of the proposed system span educational, industrial, and research contexts. In education, it allows students to explore how statistical thresholds drive technique selection on real data. In industry, it generates justified traces of each transformation decision for data-quality audits and to support regulatory compliance. In research, it provides a reproducible, ontology-based protocol for comparing preprocessing pipelines across studies. The RAG pipeline's natural-language interface further provides a natural-language interface for non-specialist professionals who need to understand the transformation rationale without having to interpret SWRL syntax or Python code

Conclusions

This work has demonstrated the feasibility of integrating an OWL ontology with parameterized SWRL rules into a RAG pipeline to support the selection of data preprocessing techniques. The ontology, designed in Protégé and comprising 47 classes, 42 individuals, and 19 SWRL rules parameterized via data properties, provides a formal and reusable specification of expert knowledge in data preprocessing. Validation on the Netflix Movies and TV Shows dataset (8,807 records, 12 columns) shows that the Python-implemented inference engine, which parses the SWRL rule structure from the RDF/XML serialization and evaluates antecedents through deterministic conditionals, without invoking a native OWL-DL reasoner, generated 15 preprocessing-technique recommendations across 9 analyzable columns. The remaining 3 columns were excluded: *show_id* as an identifier and *title* and *description* as free-text fields. Seven columns exhibited at least one data-quality problem exceeding predefined thresholds, while *date_added* and *rating* received *type-driven* or *low-nullity* recommendations. The system does not eliminate expert judgment; rather, it relocates it into the rule thresholds and makes it explicit, traceable, and reusable.

The RAG pipeline integrates four main components: (i) semantic fragmentation of the ontology into 337 knowledge units; (ii) vectorization using the multilingual *paraphrase-multilingual-MiniLM-L12-v2* model; (iii) retrieval by cosine similarity with top-k fragments ($k = 6$); and (iv) contextualized response generation using a local LLM *llama3.2:1b*, run via *Ollama*. The results show that the hybrid system generates natural-language responses consistent with ontological output, such as recommending Mode Imputation and Frequency-Based Binning for the director column, which contains 29.9% nulls and 4,528 unique values, or sequentially identifying IQR-Based Outlier Detection, Winsorization, and Robust Scaling for *release_year*, which has 8.2% outliers. The operation is performed locally on the LLM, ensuring data privacy and system reproducibility in organizational environments with connectivity restrictions.

The main contributions of this research are: (i) a reusable ontology that formalizes preprocessing expertise through parameterized rules with statistical metrics calculated dynamically from the dataset; (ii) an automatic inference engine that supports preprocessing technique selection through explicit, threshold-driven criteria, ensuring consistency for identical statistical profiles; and (iii) a RAG pipeline that provides a natural-language interface over the formal ontological output, presenting documented, traceable preprocessing plans without requiring direct interpretation of SWRL syntax. This study has several limitations: (i) the inference engine is Python-based and does not invoke a native OWL-DL reasoner; (ii) mixed-

format columns require manual type correction; (iii) the 1-billion-parameter LLM produces less cohesive text than larger models, though it enables fully offline operation; (iv) rule thresholds encode expert heuristics without empirical calibration; and (v) evaluation was limited to a single dataset without an independent ground-truth corpus or standard retrieval metrics. Future work includes: (i) integrating a native OWL-DL reasoner and comparing the Python engine's outputs through formal reasoning; (ii) validating on multiple heterogeneous datasets with objective quality metrics; (iii) conducting an ablation study of the RAG component with ground-truth queries and standard retrieval metrics, including threshold calibration via sensitivity analysis; (iv) extending SWRL rules through decision-tree learning; and (v) developing a web interface for documented preprocessing plans.

Acknowledgements and Declarations

This work was partially supported by the Vice-Rectorate of Research at Universidad del Azuay through financial and academic support as well as by staff of the Computer Science Research & Development Laboratory (LIDI).

The authors declare equal contribution and sharing of authorship roles for this publication.

The authors declare that no artificial intelligence tools have been used for the preparation of this article.

References

- Aggarwal, C. C. (2016). An introduction to outlier analysis. In *Outlier analysis* (pp. 1–34). Springer. https://doi.org/10.1007/978-3-319-47578-3_1
- Alavi, M., Leidner, D., & Mousavi, R. (2024). Knowledge management perspective of generative artificial intelligence (GenAI). *Journal of the Association for Information Systems*, 25(1), 1–12. <https://doi.org/10.17705/1jais.00859>
- Batini, C., & Scannapieco, M. (2016). Data and information quality. In *Cham, Switzerland: Springer International Publishing*. Springer International Publishing. <https://doi.org/10.1007/978-3-319-24106-7>
- Batista, G., & Monard, M. (2002). A study of k-nearest neighbour as an imputation method. In A. Abraham, J. Ruiz-del-Solar, & M. Köppen (Eds.), *Soft Computing Systems: Design, Management and Applications IOS Press* (pp. 251–260). IOS Press.
- Brys, G., Hubert, M., & Struyf, A. (2004). A robust measure of skewness. *Journal of Computational and Graphical Statistics*, 13(4), 996–1017. <https://doi.org/10.1198/106186004X12632>
- Buehler, M. J. (2024). Generative retrieval-augmented ontologic graph and multiagent strategies for interpretive large language model-based materials design. *ACS Engineering Au*, 4(2), 241–277. <https://doi.org/10.1021/acsengineeringau.3c00058>
- Cannataro, M., Comito, C., & others. (2003). A data mining ontology for grid programming. *Proc. 1st Int. Workshop on Semantics in Peer-to-Peer and Grid Computing*, 113–134. <https://scalab.dimes.unical.it/papers/pdf/SemPGRID2003.pdf>
- Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16, 321–357. <https://doi.org/10.1613/jair.953>
- Dumontier, M., Baker, C. J. O., Baran, J., Callahan, A., Chepelev, L., Cruz-Toledo, J., Del Rio, N. R., Duck, G., Furlong, L. I., Keath, N., & others. (2014). The SemanticScience Integrated Ontology (SIO) for biomedical research and knowledge discovery. *Journal of Biomedical Semantics*, 5(1), 14. <https://doi.org/10.1186/2041-1480-5-14>

- Edge, D., Krzyzanowski, T., Basisty, F., & others. (2024). A Graph RAG Approach to Query-Focused Summarization. *ArXiv Preprint ArXiv:2404.16130*. <https://doi.org/10.48550/arXiv.2404.16130>
- Fernandez-Lopez, M., & Corcho, O. (2010). *Ontological Engineering: with examples from the areas of Knowledge Management, e-Commerce and the Semantic Web*. Springer Publishing Company, Incorporated. <https://doi.org/10.1007/b97353>
- Ghosh, D., & Vogt, A. (2012). Outliers: An evaluation of methodologies. *Joint Statistical Meetings*, 12(1), 3455–3460. http://www.asasrms.org/Proceedings/y2012/Files/304068_72402.pdf
- Jaziri, W., & Sassi, N. (2026). Ontology Quality Improvement in the Semantic Web: Evidence from Educational Knowledge Graphs. *Systems*, 14(2), 154. <https://doi.org/10.3390/systems14020154>
- Krötzsch, M., Rudolph, S., & Hitzler, P. (2013). Complexities of Horn Description Logics. *ACM Transactions on Computational Logic*, 14(1), 1–36. <https://doi.org/10.1145/2422085.2422087>
- Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W., Rocktäschel, T., & others. (2020). Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33, 9459–9474. https://proceedings.neurips.cc/paper_files/paper/2020/file/6b493230205f780e1bc26945df7481e5-Paper.pdf
- Little, R. J. A., & Rubin, D. B. (2019). *Statistical analysis with missing data*. John Wiley & Sons. <https://doi.org/10.1002/9781119482260>
- McGuinness, D. L., Van Harmelen, F., & others. (2004). OWL web ontology language overview. *W3C Recommendation*, 10(10), 2004.
- Meyer, J. G., Urbanowicz, R. J., Martin, P. C. N., O'Connor, K., Li, R., Peng, P.-C., Bright, T. J., Tatonetti, N., Won, K. J., Gonzalez-Hernandez, G., & others. (2023). ChatGPT and large language models in academia: opportunities and challenges. *BioData Mining*, 16(1), 20. <https://doi.org/10.1186/s13040-023-00339-9>
- Páez, A., & Boisjoly, G. (2022). *Exploratory Data Analysis* (pp. 25–64). https://doi.org/10.1007/978-3-031-20719-8_2
- Paltin Chica, D. L., Mejía Mendieta, J. D., Orellana, M., & Zambrano-Martinez, J. L. (2025). Modelado Semántico de Emergencias del ECU 911 con NLP y Ontologías. *Revista Tecnológica - ESPOL*, 37(E1), 112–125. <https://doi.org/10.37815/rte.v37nE1.1351>
- Park, Y., Witherell, P., Surovi, N. A., & Cho, H. (2024). Ontology-based Retrieval Augmented Generation (RAG) for GenAI-supported Additive Manufacturing. *The 35th Annual International Solid Freeform Fabrication Symposium*. https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=958349
- Publio, G. C., Esteves, D., Panov, P., Soldatova, L., Soru, T., Vanschoren, J., Zafar, H., & others. (2018). ML-schema: exposing the semantics of machine learning with schemas and ontologies. *ArXiv Preprint ArXiv:1807.05351*. <https://doi.org/10.48550/arXiv.1807.05351>
- Rattanasawad, T., Saikaew, K. R., Buranarach, M., & Supnithi, T. (2013). A review and comparison of rule languages and rule-based inference engines for the Semantic Web. *2013 International Computer Science and Engineering Conference (ICSEC)*, 1–6. <https://doi.org/10.1109/ICSEC.2013.6694743>
- Reimers, N., & Gurevych, I. (2019, November). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. <http://arxiv.org/abs/1908.10084>
- Sharma, K., Kumar, P., & Li, Y. (2025). OG-RAG: ontology-grounded retrieval-augmented generation for large language models. *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, 32950–32969. <https://doi.org/10.18653/v1/2025.emnlp-main.1674>
- Yang, Y., & Webb, G. I. (2009). Discretization for naive-Bayes learning: managing discretization bias and variance. *Machine Learning*, 74(1), 39–74. <https://doi.org/10.1007/s10994-008-5083-5>