Assessing Ontology Quality With QASAR

Gonzalo Nicolás-Martínez[1], Francisco Abad-Navarro[1], Manuel Quesada-Martínez[2], Astrid Duque-Ramos[1,3], Belén Juanes-Cortés[1] and Jesualdo Tomás Fernández-Breis[1,*]

[1]*Departamento de Informática y Sistemas,Universidad de Murcia, CEIR Campus Mare Nostrum, IMIB-Pascual Parrilla, Murcia, España*

[2]*Universidad Miguel Hernández, Elche, Alicante, España*

[3]*Grupo Ingeniería y Tecnologías de las Organizaciones y de la Sociedad (ITOS), Departamento de Ingenería de Sistemas, Facultad de Ingeniería, Universidad de Antioquia, A.A 1226, Medellín, Colombia.*

Abstract

Ontologies are foundational for enabling semantic interoperability, data integration, and knowledge sharing across a broad spectrum of domains, including biomedicine, engineering, and the social sciences. As their adoption continues to grow, so too does the need for robust methods to ensure their quality. Despite the development of a range of quality assurance frameworks and methodologies aimed at evaluating various aspects of ontology quality, the practical application of these methods remains fragmented. In this demonstration, we present QASAR, a novel, integrated quality assurance tool designed to bridge this gap. QASAR consolidates several QA methodologies developed by our team into a unified, user-friendly platform. The system supports a broad set of quality dimensions and alignment with best practices and community standards. By offering automated analyses, interactive reports, and guided recommendations, QASAR enables both novice and experienced ontology developers to iteratively improve their ontologies with greater ease and confidence. Through this tool, we aim to promote the creation of higher-quality ontologies and foster broader adoption of quality assurance practices within the ontology engineering lifecycle. QASAR represents a step forward in operationalizing ontology quality assurance by providing a streamlined and accessible environment for comprehensive ontology evaluation.

Keywords

Knowledge representation, ontologies, quality assurance

1. Introduction

Semantic resources, such as ontologies, have become essential components for facilitating interoperability and data sharing across a wide range of domains, including biomedicine, engineering, and the social sciences. As reliance on these resources grows, ensuring their quality has become increasingly critical. However, despite the recognized importance of ontology quality assurance (QA), there remains a notable lack of integrated tools to effectively support ontology developers in this process. In recent years, the ontology engineering community has devoted considerable effort to defining QA methodologies and establishing standards aimed at improving various aspects of semantic resources. These approaches typically target different dimensions of quality—such as structural correctness, logical consistency, completeness, and domain relevance—and often propose diverse strategies for evaluating these aspects in a quantitative and reproducible manner. Nonetheless, the resulting QA methods are often implemented in isolated tools or frameworks, making it challenging for developers to apply them cohesively within a unified workflow. This fragmentation poses a significant challenge, especially for ontology developers who require comprehensive, practical support in maintaining high-quality ontologies. While individual QA tools exist, few offer the integration of multiple, rigorously tested methods within a single, user-friendly environment. To address this gap, we present QASAR, an ontology QA tool designed to consolidate various quality frameworks developed by our research group into a unified

Proceedings of FOIS 2025 Satellite events co-located with the 15th International Conference on Formal Ontology in Information Systems (FOIS 2025), September 10-12, 2025, Catania, Italy

*Corresponding author.

gonzalo.nicolasm@um.es (G. Nicolás-Martínez); francisco.abad@um.es (F. Abad-Navarro); mquesada@umh.es (M. Quesada-Martínez); astrid.duquer@udea.edu.co (A. Duque-Ramos); belen.juanesc@um.es (B. Juanes-Cortés); jfernand@um.es (J. T. Fernández-Breis)

platform. QASAR provides an end-to-end quality assurance workflow that guides developers through the evaluation and improvement of their semantic resources. By integrating diverse quality metrics and offering actionable feedback, QASAR supports both novice and expert users in the systematic development and maintenance of high-quality ontologies.

2. The QASAR Framework

QASAR provides qualitative and quantitative ontology information that allows developers and researchers to evaluate their semantic resources, assessing their strengths and weaknesses through the use of four ontology quality-related frameworks, namely, OQuaRE [1], OntoEnrich [2], HURON [3], and Evaluome [4]. OQuaRE provides the quality model, which is structured into quality characteristics that are subsequently divided into quality sub-characteristics, which are then measured through quality metrics. OntoEnrich provides additional quality metrics based on the lexical regularities exhibited by the labels of the entities, as well as allowing users to perform lexical analyses over their resources. HURON provides quantitative metrics for assessing the human readability of ontologies and for evaluating the adherence to best practices. Finally, Evaluome provides corpus analysis capabilities through its clustering-based data analysis. QASAR's information is represented using the Ontology QUality Ontology (OQUO)[1], which is a modular ontology that contains the semantic definitions needed to represent quality models based on metrics, characteristics, and subcharacteristics, as well as the entities required to represent ontology evaluations. QASAR is designed not only to enable users to assess the quality of their semantic resources but also to support their improvement through guided, informed modifications. As users make changes to their ontologies, QASAR continuously tracks and records the quality metrics, allowing them to monitor how their interventions affect the overall quality over time. The tool proactively identifies existing issues within a resource by pinpointing their causes, specific locations, while also assessing their severity and potential impact on the ontology's quality. This diagnostic capability empowers users to detect flaws that might otherwise go unnoticed.

2.1. OQuaRE

The Ontology Quality and Requirement Evaluation framework (OQuaRE) [1] adapts the ISO/IEC 25000:2005 standard (SQuaRE) for software product quality to the specific context of ontology evaluation.

The **Quality model** provides a set of high-level quality characteristics and corresponding subcharacteristics. They serve as conceptual dimensions for evaluating ontology quality. The OQuaRE quality characteristics are:

- Compatibility: The capability of two or more ontology components to exchange information and/or perform their required functions while sharing the same hardware or software environment.
- Functional adequacy: The capability of the ontologies to provide concrete functions.
- Maintainability: The capability of ontologies to be modified in response to changes in environments, requirements or functional specifications.
- Operability: Effort needed for use, and on the individual assessment of such use, by a stated or implied set of users.
- Quality in use: The degree to which a product used, by specific users, meets their needs to achieve specific goals.
- Reliability: Capability of an ontology to maintain its level of performance under stated conditions for a given period of time.
- Structural: Assessment of structural aspects of the ontology such as consistency, formalisation, redundancy or tangledness.

[1]https://github.com/tecnomod-um/oquo

- Transferability: The degree to which the ontology can be transferred from one environment to another.

The **Quality measurement** division includes both basic metrics, which can be directly extracted from the ontology (e.g., the number of classes, properties, or axioms), and derived metrics, which are calculated from the basic ones. The metric values are mapped to a 1–5 scale according to predefined thresholds.

The **Quality evaluation** component assesses whether the ontology meets its design requirements, that is, the expected features for the developers of the ontology, For this purpose, we aggregate the metric values through their associated subcharacteristics and characteristics. In OQuaRE, each subcharacteristic is linked to a single quality characteristic, while metrics may contribute to multiple subcharacteristics. The associations between metrics, subcharacteristics, and characteristics are available at https://github.com/tecnomod-um/oquare.

2.2. HURON

HURON [3] provides a set of quantitative metrics focused on the human-readable content of ontologies and supports the analysis of the application of best practices. In particular, HURON provides metrics related to the ratio of names, synomyms, and descriptions per class, object property, datatype property, and annotation property as well as metrics that measure the percentage of entities lacking names, synonyms or descriptions. HURON assesses the following best practices: (1) classes must have one canonical name; (2) classes must contain one description; (3) classes must include as many synonyms as possible; (4) class names must define the concept represented as clearly as possible, using a systematic nomenclature; and (5) Application of the *lexically suggest, logically define (LSLD)* principle [5], which means that there should be a relation between the axioms and the natural language content.

2.3. OntoEnrich

The OntoEnrich framework [2] comprises methods for the automatic detection and analysis of lexical regularities. A lexical regularity is a group of consecutive tokens that appear in several labels of the ontology classes. For example, "transcription pathway" is a lexical regularity that appears in 13 classes of the Pathway Ontology. An analysis guided by these regularities allows the exploration of different lexical/semantic aspects. The analysis of lexical regularities permits the identification of potentially missing or wrong axioms by comparing the lexical information and the logical axioms. It provides metrics to: (1) analyse how a given lexical regularity is distributed along the ontology modules (modularity); (2) analyse how a given lexical regularity is exhibited in a given context of the ontology (locality); and (3) identify matches of lexical regularities with the labels of classes in the same ontology (intra) or with external ontologies (extra) through cross-product extension metrics (CPE).

2.4. Evaluome

This framework provides information about the partitioning of the dataset generated by the quality metrics. It quantifies two statistical properties of the clusterings, namely stability and quality of the clustering, by computing different indexes on sequential data partitions, such as the Jaccard or Silhouette width. On the one hand, stability reports whether a cluster is subject to change by small variations, whereas the quality or goodness of clusters refers to how closely related the individuals in a cluster are, and how well the resulting clusters are separated from each other. Both cluster indexes are used to propose the number of clusters that provides the best partitioning of the datasets.

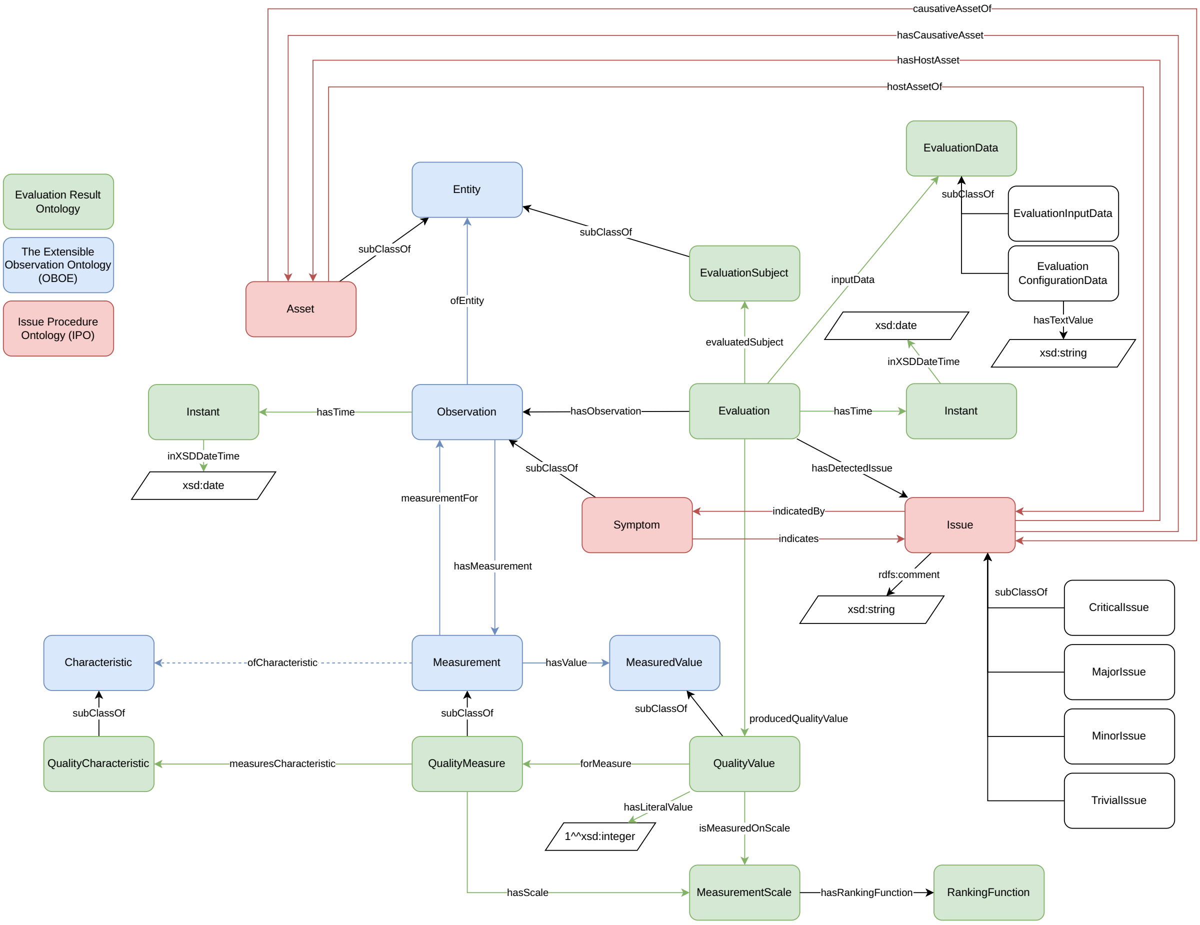

Figure 1: Graphical representation of the OQUO ontology

2.5. OQUO: Ontology QUality Ontology

The QASAR quality model is defined by the Ontology QUality Ontology (OQUO)[2], which is described as *a modular ontology that contains the semantic definitions needed to represent quality models based on metrics, characteristics, and sub-characteristics, as well as the entities needed to represent ontology evaluations*. OQUO provides (1) A quality model in which the quality characteristics to be measured in an ontology are linked to their corresponding quality metrics; (2) An observation result model, with concepts and relationships to represent observations, measurements, evaluations, issues, scales, etc; and (3) A common vocabulary within quality frameworks to evaluate ontologies, allowing their corresponding tools to normalize their outputs to a single, shared model. OQUO reuses and aligns the Evaluation Result Ontology (EVAL)[3], the Extensible Observation Ontology (OBOE) [6], and the Issue Procedure Ontology (IPO) [7], as shown in Figure 1. Since QASAR uses OQUO, the metrics and outputs from OQuaRE, HURON, and OntoEnrich are described with it, making their results interoperable. OQUO also enables users to define their own metrics so they can be consumed by QASAR.

3. The QASAR Tool

QASAR is currently available at https://semantics.inf.um.es/qasar. Next, we describe the main workflow for using QASAR. The user provides one or more ontologies in any valid OWL format supported by QASAR, such as Turtle or RDF/XML. The user can either select files stored locally on their PC,

[2]https://github.com/tecnomod-um/oquo
[3]http://purl.org/net/EvaluationResult

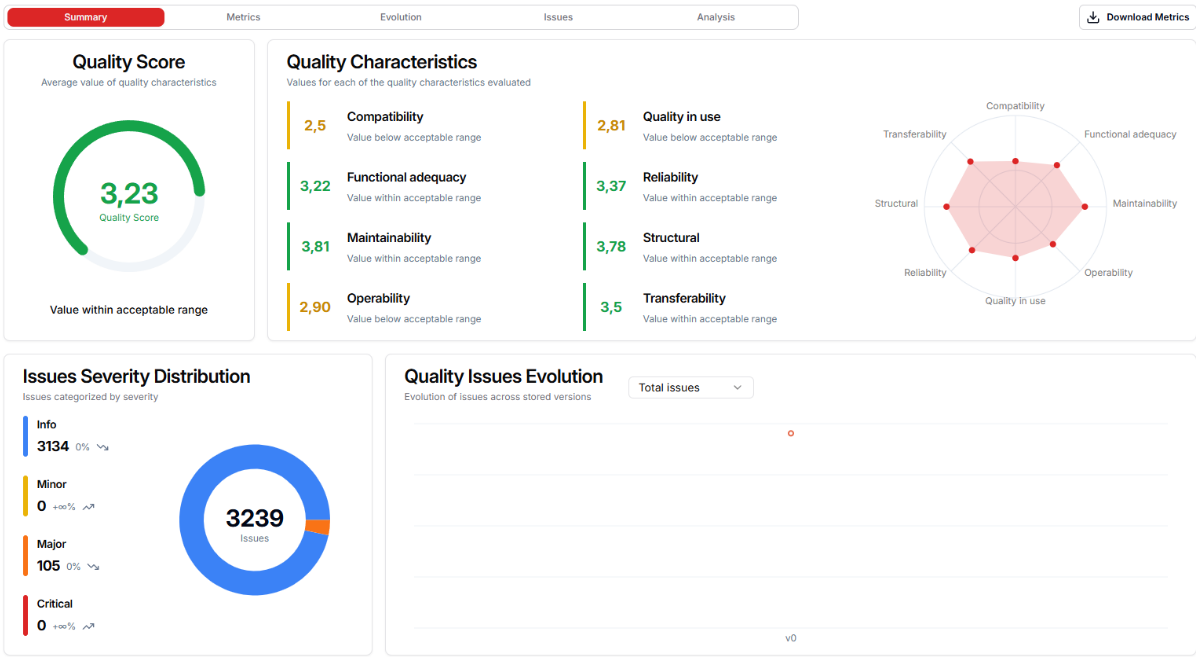

Figure 2: Quality summary provided for a particular ontology, with colors representing the following ranges of values in the 1-5 scale: red for [1-2) (poor), yellow for [2, 3) (below acceptable), green for [3, 5) (good), and blue for 5 (optimal).

or specify an ontology IRI. QASAR can also retrieve ontologies from GitHub repositories. Once the ontologies are uploaded and validated, a task is created and queued to obtain the metrics from OQuaRE, HURON, and OntoEnrich, which are generated in OQUO-compliant RDF. An example of this RDF is available at https://shorturl.at/GlJvK. The generated output is stored in the QASAR databases, and the different services are updated with the results. QASAR is now able to provide the user with the different calculated metrics, quality characteristics, and sub-characteristics as well as present the issues detected during the analysis. These results can be explored at the level of a single ontology or at the level of an ontology corpus, in case the project includes more than one ontology. Both levels of analysis are described in the following subsections. QASAR also offers a read-only demo user, which enables access to the results of a specific corpus of ontologies.

3.1. Ontology Analysis

QASAR provides a set of views that users can navigate through when checking the results obtained for a semantic resource. Each of the views presents different information regarding the quality of the resource and can be used for different purposes when it comes to performing modifications and improvements on it.

The summary view (see figure 2) displays the different values obtained for each quality characteristic, together with a visual representation to quickly gauge the general quality of the resource. It also presents a general score obtained from the values of each quality characteristic, which are not really representative of its actual quality, but provide a surface level idea of whether it is reaching the quality goals defined by QASAR. In addition, it provides information on the number of issues detected for the current version of the resource, as well as its evolution, which is helpful for evaluating the impact of the changes performed. The issues are classified by severity and shown in different colours: Info (blue) for non-impactful issues requiring no action, Minor (yellow) for low impact, Major (orange) for significant impact, and Critical (red) for high-impact issues requiring immediate action.

The detailed report generated for a semantic resource includes the values of all the quality characteristics and sub-characteristics, as well as the raw metrics used to calculate them. Each characteristic and

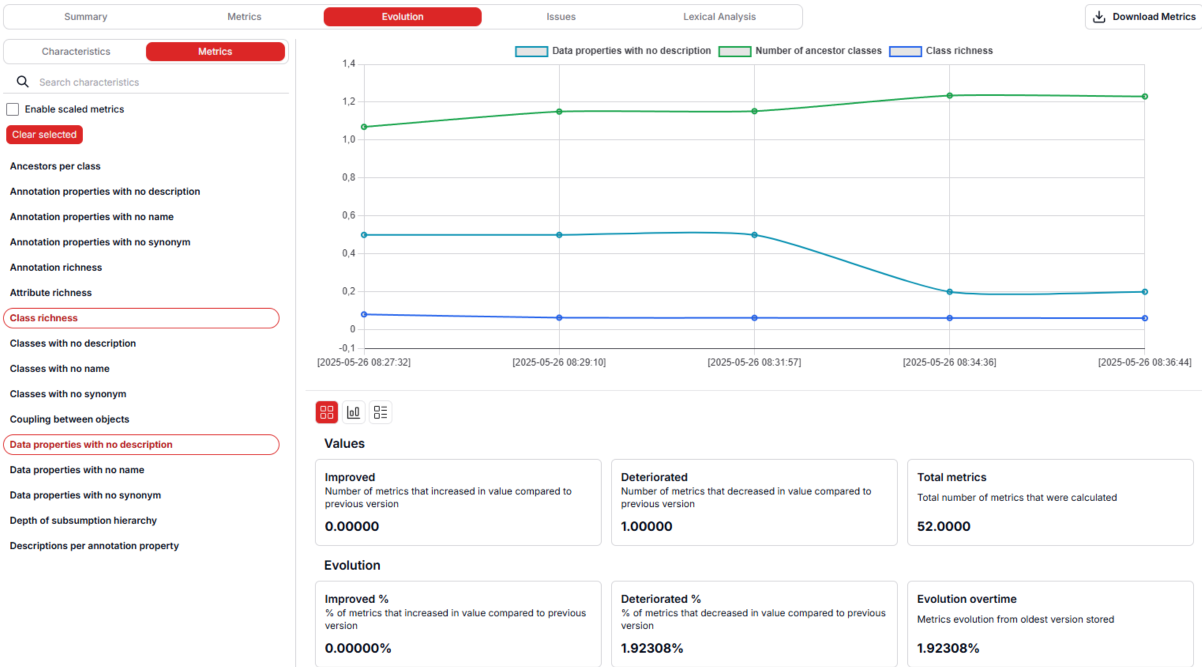

Figure 3: Evolution of metrics for a particular ontology

metric is accompanied by its own definition, raw values and, when applicable, scaled values or scores, along with an explanation of how it influences or is influenced by other metrics and characteristics.

QASAR tracks the progression of each resource across multiple ontology versions by analysing the values of its quality metrics, characteristics, and subcharacteristics, as shown in Figure 3. Users can explore how these values have evolved over the latest 20 versions and dynamically display detailed information for the specific metrics or characteristics they are interested in.

Users can configure thresholds for the available quality metrics (**Rulesets**). When a metric falls within a defined threshold, QASAR generates an issue for the resource, which is useful to properly control whether certain values for specific metrics are met. A rule can have multiple thresholds for a single metric, tied to the different severities available. Each issue is also classified with a severity level, which indicates how it affects the quality of the resource. The severity grading goes from *'Info'* level, whose effect on quality is minimal and mostly informative, to *'Critical'*, which indicates an extremely important flaw that should be fixed immediately, as its effect on the quality of the resource is such that it renders the resource unusable. Another classification applied distinguishes whether the issue has been detected at a *'Class'* level, which comes from the QASAR quality frameworks, or *'Ontology'* level, which is a flaw detected at the resource level, indicated by user-defined rulesets. Figure 4 shows how users can explore a list of issues detected by the quality frameworks integrated into QASAR. Each issue listed provides detailed information, such as the IRI of the associated entity, a description paired with a set of recommended steps to properly address the issue, and the possible associated entities that are part of the issue along with the main entity.

Figure 5 shows the view that allows the list of lexical regularities to be explored. The bottom right shows a sortable list of regularities, while the left side offers access to analysis settings and metric calculations. Users can click on a regularity (e.g., "transcription pathway") to view detailed metrics. Figure 6 shows the expanded view of the lexical regularity "transcription pathway". The different metrics provide concrete information about this regularity. For example, it can be seen that all the classes exhibiting this regularity are descendants of the "pathway" and "transcription pathway" classes of interest, which indicates the application of systematic naming. An issue is generated for the classes that do not follow a systematic naming. The locality value means that not all classes exhibiting regularity are siblings, but rather they span more than one hierarchical level.

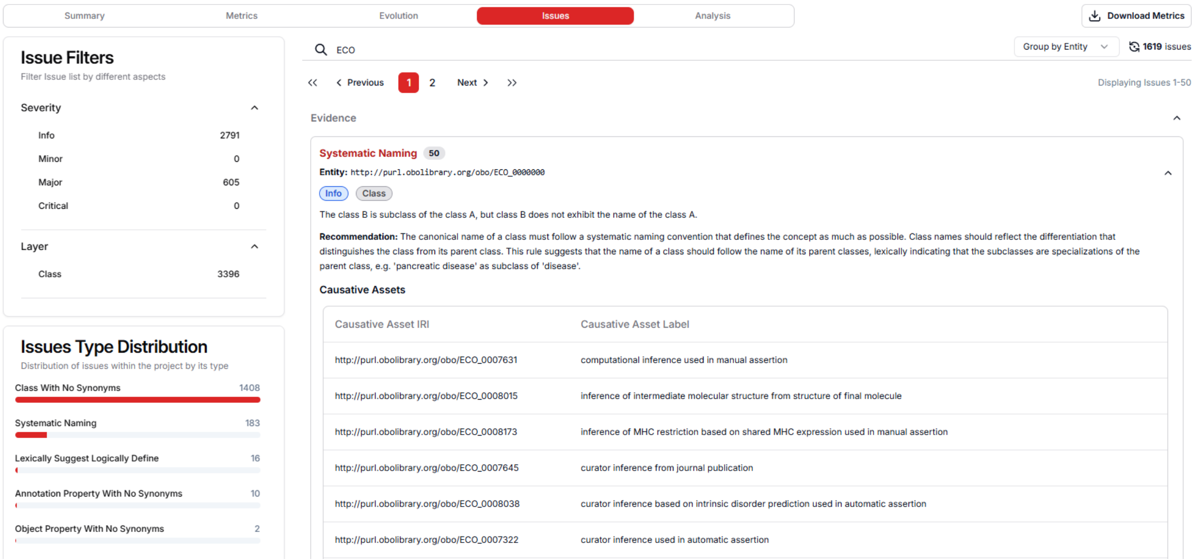

Figure 4: View of issues detected within a semantic resource

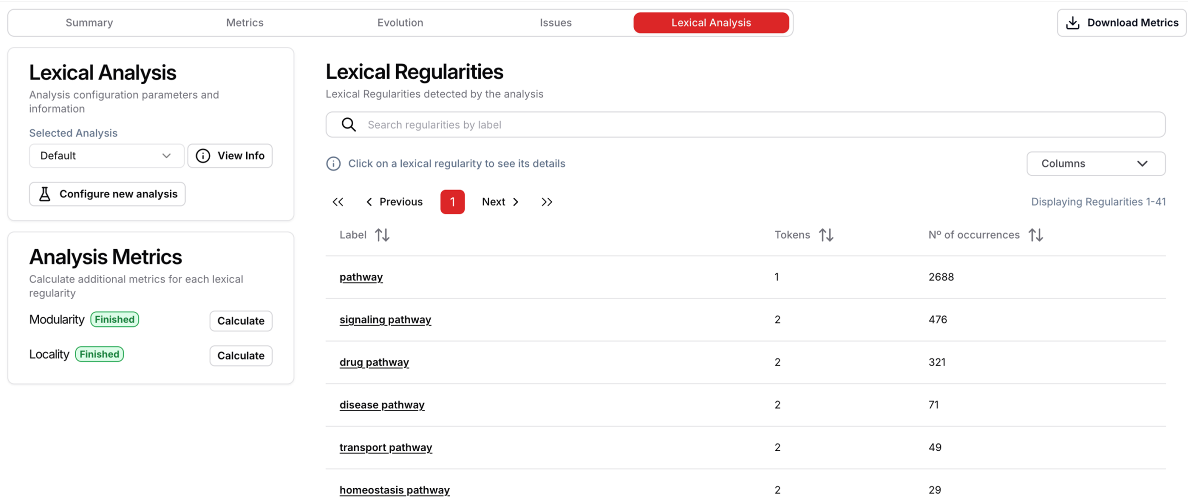

Figure 5: View of a lexical analysis of the Pathway Ontology

3.2. Corpus Analysis

The project summary (Figure 7) provides average values for each quality characteristic and includes a dashboard for quick insights on the corpus. An issues view displays the distribution and quantity of detected issues across ontologies. Additionally, QASAR integrates Evaluome to assess the reliability and behavior of quality metrics, identifying patterns and outliers. The results are displayed as density charts, as shown in figure 8, where the user can dynamically change the number of clusters that are made, together with the optimal number of clusters identified during the analysis. Each possible cluster choice come with two different metrics, namely *Quality* and *Stability*. Quality assesses how similar an instance is to other instances within the same cluster and how dissimilar it is to the rest.

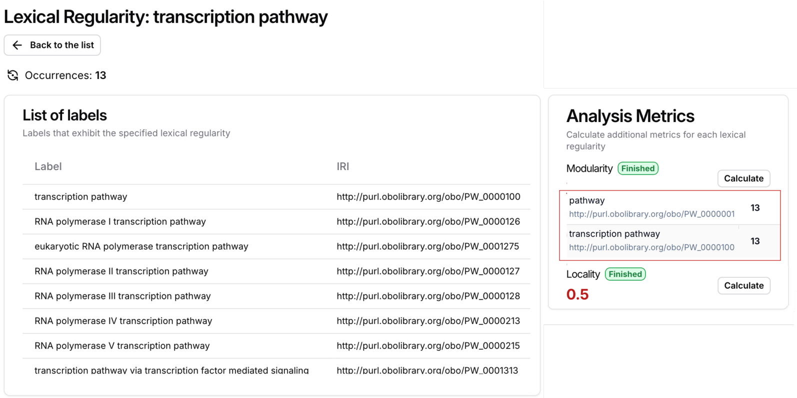

Figure 6: Analysis driven by the lexical regularity “transcription pathway

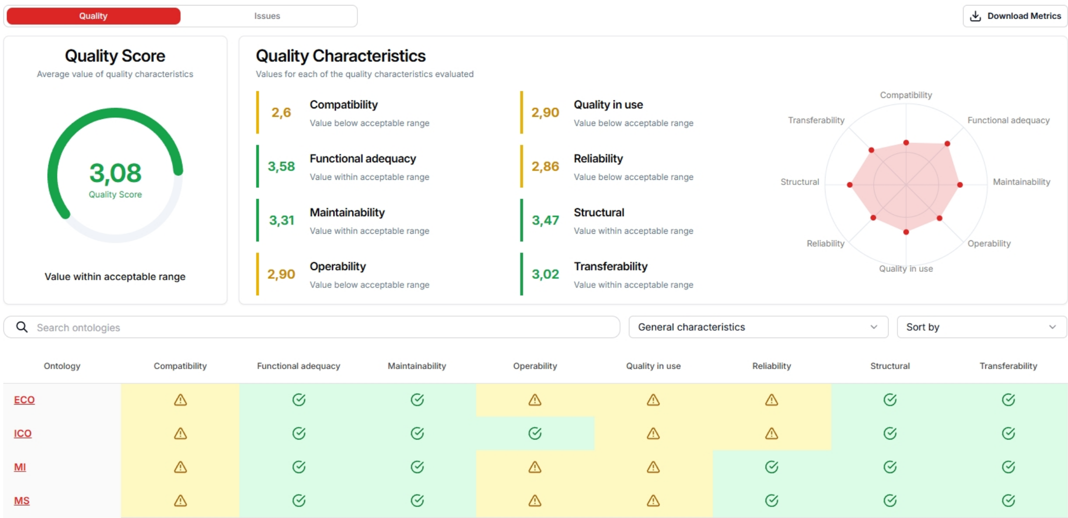

Figure 7: Quality summary provided for a project, displaying information about its ontologies

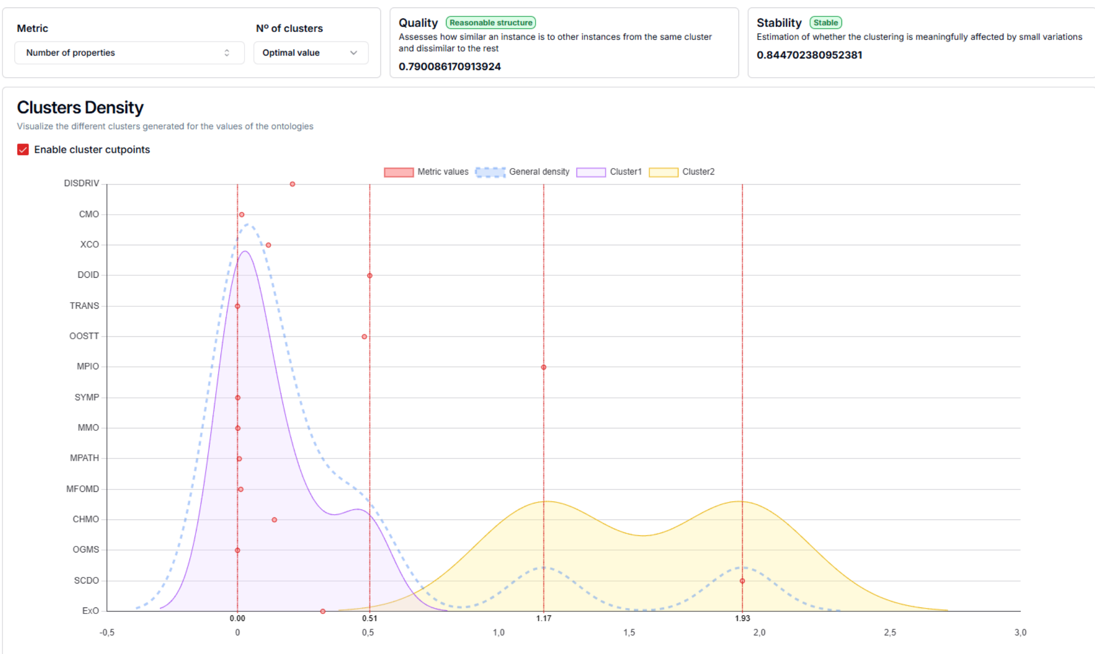

Figure 8: Quality summary provided for a project, displaying information about its ontologies

Conclusions

In this paper, we have described QASAR, a tool for the quality assurance of semantic resources. The current implementation integrates a series of frameworks developed by our research group, but due to the use of OQUO, it could be easily adapted to integrate other quality dimensions and metrics.

Acknowledgments

This research has been funded by FS/10.13039/100007801 (22529/PDC/24).

Declaration on Generative AI

The authors have not employed any Generative AI tools.

References

[1] A. Duque-Ramos, J. Fernandez-Breis, R. Stevens, N. Aussenac-Gilles, OQuaRE: A SQuaRE-based approach for evaluating the quality of ontologies, Journal of Research and Practice in Information Technology 43 (2011) 159–176.
[2] M. Quesada-Martínez, J. T. Fernández-Breis, R. Stevens, N. Aussenac-Gilles, Ontoenrich: A platform for the lexical analysis of ontologies, in: International conference on knowledge engineering and knowledge management, Springer, 2014, pp. 172–176.
[3] F. Abad-Navarro, C. Martínez-Costa, J. T. Fernández-Breis, Huron: A quantitative framework for assessing human readability in ontologies, IEEE Access 11 (2023) 101833–101851.
[4] J. A. Bernabé-Díaz, M. Franco, J.-M. Vivo, M. Quesada-Martínez, J. T. Fernández-Breis, An automated process for supporting decisions in clustering-based data analysis, Computer Methods and Programs in Biomedicine 219 (2022) 106765.
[5] A. Rector, L. Iannone, Lexically suggest, logically define: quality assurance of the use of qualifiers and expected results of post-coordination in SNOMED CT, Journal of biomedical informatics 45 (2012) 199–209.
[6] J. Madin, S. Bowers, M. Schildhauer, S. Krivov, D. Pennington, F. Villa, An ontology for describing and synthesizing ecological observation data, Ecological informatics 2 (2007) 279–296.
[7] M. D. d. Morais, M. D. d. A. Jacyntho, Issue procedure ontology (IPO): uma ontologia para sintomas, problemas e soluções, Perspectivas em Ciência da Informação 21 (2017) 03–28.