Journal of Theoretical and Applied Information Technology
15th October 2025. Vol.103. No.19

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

INTEGRATING COMPETENCY QUESTION-DRIVEN ONTOLOGY AUTHORING WITH ONTOLOGY VALIDATION FOR LEARNING ANALYTICS FROM HETEROGENEOUS PLATFORMS

SYAHMIE SHABARUDIN[1], SAZILAH SALAM[1], IBRAHIM AHMAD[1], MOHD HAFIZAN MUSA[1], NURFADHLINA MOHD SHAREF[2], DALBIR SINGH[3], WENDY HALL[4], ROHANA MAHMUD[5]

[1]Fakulti Teknologi Maklumat dan Komunikasi, Universiti Teknikal Malaysia Melaka,Melaka, Malaysia

[2]Faculty Of Computer Science and Information Technology, Universiti Putra Malaysia, Selangor, Malaysia

[3]Center for Software Technology and Management (SOFTAM), Faculty of Information Science & Technology, Universiti Kebangsaan Malaysia, Selangor, Malaysia

[4]University of Southampton, United Kingdom

[5]Universiti Malaya, Wilayah Persekutuan Kuala Lumpur, Malaysia

E-mail: [1]p032110014@student.utem.edu.my, [1]sazilah@utem.edu.my, [1]ibrahim@utem.edu.my, [2]p032220014@student.utem.edu.my, [2] nurfadhlina@upm.edu.my, [3]dalbir@ukm.edu.my, [4]wh@ecs.soton.ac.uk,

ABSTRACT

Ontology authoring for learning analytics from heterogeneous platforms is a complex task, particularly for authors who may lack proficiency in logic. This paper introduces a novel approach that leverages competency questions (CQs) and test-driven development principles to streamline ontology, authoring and validation. We analyse common questions from stakeholders at 13 public universities to create competency questions, identify patterns, and utilise linguistic presuppositions to define ontology requirements. Our methodology ensures that these requirements are testable and can be validated, facilitating an integrated ontology for learning analytics. Additionally, we present a detailed ontology validation report, demonstrating the effectiveness of our approach through consistency checks, property validations, and individual test cases. This integrated method aims to enhance the accuracy and reliability of ontologies in representing learning analytics data from diverse platforms.

Keywords: *Competency Questions, Ontology Validation, Learning Analytics, Heterogeneous Platforms, Test-Driven Ontology Development*

1. INTRODUCTION

The increasing volume and diversity of educational data from various digital platforms present both significant opportunities and complex challenges for advanced learning analytics. While numerous studies have highlighted the immense potential of learning analytics in enhancing educational processes and outcomes, the heterogeneous sources and formats of this data often hinder its effective integration and utilization. In response, ontologies, which provide structured representations of knowledge, have emerged as a powerful and widely adopted solution to unify and leverage these disparate datasets [1, 7]. They enable a common understanding of data across platforms, facilitating accurate educational data analysis and supporting interoperability, standardization, and effective decision-making in educational settings [2, 4].

Previous research has demonstrated the instrumental role of ontologies in designing and developing robust learning analytics systems by integrating learning design and content through structured classes and relationships [5]. Ontology-based metadata models have also proven effective in providing description specifications and search tools for integrating various forms of educational resources [6]. Furthermore, ontologies aid in determining the classes, entities, and properties to include in databases for effective data integration [8] and support the construction of educational and career-oriented recommendation systems, enhancing personalized learning experiences [9].

Despite these established benefits and the widespread use of ontologies in data integration, the inherent complexities associated with their authoring and rigorous validation remain significant challenges. This study builds upon the foundational understanding of ontologies in learning analytics by addressing these specific complexities, particularly for domain experts lacking formal training in ontology design.

This study aims to address these critical challenges (as detailed in Section 1.1) by introducing a novel approach that integrates competency question-driven ontology authoring (CQ-DOA) with comprehensive, test-driven ontology validation techniques. Our motivation stems from the need to streamline the ontology development process, making it more accessible to domain experts and ensuring the resulting ontologies are robust, consistent, and directly applicable to real-world learning analytics scenarios.

Based on these challenges and motivations, this paper seeks to answer the following research questions:

1. How can common questions from educational stakeholders be systematically collected, analysed, and translated into formal competency questions to define the functional requirements of an ontology for learning analytics?
2. How can an integrated ontology for learning analytics be developed that effectively captures patterns identified through competency questions and unifies data from heterogeneous platforms?
3. How can a robust validation framework, employing test-driven development principles, be established and applied to automatically test competency questions against the ontology, ensuring its accuracy and reliability?
4. How can the practical applicability and effectiveness of the developed ontology and validation framework be demonstrated through detailed case studies and real-world data integration scenarios?

We hypothesize that by systematically leveraging competency questions and applying test-driven development principles, we can create a more accurate, consistent, and practically applicable ontology for learning analytics that effectively integrates and analyses data from diverse platforms, thereby overcoming the complexities inherent in traditional ontology authoring and validation.

This paper makes several key contributions to the field of learning analytics and ontology authoring:

- Competency Question Framework: We propose a systematic approach for formulating competency questions from real-world stakeholder queries, providing a practical methodology for capturing precise ontology requirements.
- Integrated Ontology Development: We present the creation of a comprehensive ontology for learning analytics designed to unify data from diverse platforms, thereby enhancing the ability to perform accurate and meaningful analyses.
- Validation Methodology: We implement a robust validation framework that employs test-driven development principles to ensure the accuracy and reliability of the ontology, moving beyond traditional, manual validation.
- Practical Evaluation: We demonstrate the practical applicability and effectiveness of the developed ontology and validation framework through detailed case studies and real-world data integration scenarios, offering valuable insights for future research and practice in educational data integration and analysis.

By addressing these contributions, this study aims to significantly enhance the usability and effectiveness of ontology authoring for learning analytics, ultimately supporting more informed, data-driven decision-making in educational environments.

Ontologies are important in addressing the challenges of integrating educational data from diverse sources for advanced learning analytics. They provide a structured framework for representing knowledge, enabling interoperability, standardisation, and effective data integration across various platforms, enhancing decision-making and strategy formulation in educational environments.

1.1 Challenges In Ontology Authoring for Learning Analytics

Ontology authoring is a complex and multifaceted task that demands a deep understanding of both the specific domain and formal logic. This inherent complexity presents several significant challenges that often impede the efficient and accurate development of ontologies, particularly within the context of learning analytics:

- **Difficulty in Requirement Specification:** Domain experts, who possess invaluable knowledge but often lack formal training in ontology design, frequently struggle to articulate ontology requirements in a manner that can be directly translated into formal axioms. This disconnect complicates the initial phases of development, making it difficult to precisely capture the intended knowledge [11].
- **Integration of Heterogeneous Data Sources:** Learning analytics data originates from diverse digital platforms, each with varied data structures and semantics. Integrating these heterogeneous datasets into a cohesive and unified ontology is a non-trivial task that necessitates careful mapping, alignment, and transformation processes [12].
- **Limitations of Traditional Validation and Testing:** Ensuring that a developed ontology accurately reflects the domain and meets specified requirements is paramount. However, conventional validation methods are often time-consuming, labour-intensive, and prone to errors, requiring extensive manual effort and specialized expertise [13]. This limits the efficiency and reliability of the validation process.
- **Usability Barriers for Non-Experts:** Many existing ontology authoring tools are primarily designed for users with a strong background in formal logic. This design paradigm significantly limits their accessibility and usability for domain experts, who are the primary stakeholders responsible for defining and verifying the accuracy and relevance of the ontology in educational settings [14].

These challenges collectively highlight the need for more streamlined, accessible, and rigorously validated approaches to ontology authoring for learning analytics, which this study aims to address.

2. COMPETENCY QUESTION-DRIVEN ONTOLOGY AUTHORING

Competency questions (CQs) are widely recognized as a core part of creating ontologies. They act as a vital link, helping to translate the practical knowledge of experts into the clear, structured format needed for building effective ontologies. CQs are important because they turn informal, everyday questions into precise requirements that computers can understand. This helps avoid confusion and makes sure the ontology does exactly what it's supposed to. Past studies have consistently shown how useful CQs are for clarifying what's needed, guiding development, making validation easier, and generally improving how ontologies are used [15]. For example, CQs are crucial for clearly stating what an ontology needs to do, making sure it lines up with what users expect and need it for [16]. They provide a solid, user-focused starting point for building ontologies step-by-step, carefully guiding which concepts, relationships, and rules are included so they are truly relevant to the subject area [17]. Also, CQs often work as built-in tests for checking the ontology, allowing developers to confirm that it accurately answers specific questions and behaves reliably in different situations [18]. Their unique ability to connect technical details with practical, real-world problems also makes ontologies much easier for non-experts to understand and use, encouraging more people to adopt and benefit from these knowledge models [19].

While CQs are clearly important and have been used in many areas, current approaches often struggle to systematically get a full set of CQs from the varied needs of different people involved. Also, smoothly integrating these CQs into a strong, test-driven validation system is still a challenge. For instance, some methods might focus too much on just the theory of CQs, without getting enough real-world input from stakeholders. This can lead to ontologies that are technically sound but not very practical or don't quite meet the specific needs of the area. On the other hand, some approaches might not have automated ways to directly check the ontology against its CQs, relying instead on manual, time-consuming validation. These manual steps are prone to human error and inefficiency, which hurts how thorough and scalable the validation process can be. Our current study builds on these existing ideas by introducing a new method. This approach not only systematically gathers and carefully analyzes common questions from a wide range of people in education making sure it's relevant to real-world use but also smartly uses linguistic presuppositions to turn those underlying needs into requirements that can be tested. This systematic way of getting requirements, combined with a complete and automated validation system, makes our study stand out. It offers a more thorough, efficient, and practical way to ensure ontologies are accurate, consistent, and reliable in the complex and varied world of modern learning analytics.

2.1 Methodology For Collecting and Analysing Common Questions

To create an ontology that truly meets the needs and perspectives of its users, it's essential to systematically gather and analyze common questions from the relevant people involved. Our method of doing this involves several clear and step-by-step phases:

i. **Identifying Stakeholders:** First, we carefully identified the key people who are actively involved in or directly affected by using and analysing learning analytics data. This diverse group included academics (like instructors and researchers), administrative staff (like department heads and program coordinators), and technical staff (like data analysts and IT support) from various educational institutions. Including people from these different roles was crucial for getting a complete picture of requirements from various operational, teaching, and analytical viewpoints.

ii. **Designing the Survey:** We then carefully designed a comprehensive survey to collect a wide range of common questions related to learning analytics. The survey included both open-ended questions, which allowed participants to share many different questions and concerns without limits, and closed-ended questions, which were structured to help us find frequently asked questions and repeated themes for analysis. In our study, the survey was organized into six distinct sections to make sure we covered all important aspects of learning analytics in today's educational settings:
   - Part A: How Learning Management Systems (LMS) are Used – This section focused on understanding how users interact with and get insights from data generated within LMS platforms.
   - Part B: End-of-Semester Reporting Needs for Student & Course Achievement – This part looked at what reports and data are needed at key academic times, like the end of a semester.
   - Part C: Ways to Monitor Students During the Semester – This section explored how student progress is tracked continuously throughout an academic term.
   - Part D: Actions Taken for Student Monitoring During the Semester – This part delved into the types of interventions, responses, and teaching adjustments made based on student monitoring data.
   - Part E: Systems for Data-Driven Monitoring During the Semester – This section examined the technical and procedural frameworks that support using data for monitoring.
   - Part F: Challenges, Opportunities, and Benefits of Student Monitoring Systems During the Semester – This final section aimed to get qualitative insights into the bigger picture, including difficulties, potential improvements, and advantages of using learning analytics.

iii. **Collecting Data:** After designing the survey, we systematically sent it to the identified stakeholders across many educational institutions. This approach was chosen to get a variety of viewpoints and to find common needs that apply across different academic settings. In this study, we successfully collected responses from a significant group of 22 participants, who together represented 13 different public universities. This strong dataset provided a rich and representative foundation for our analysis.

iv. **Analysing Questions:** The collected questions were thoroughly analysed using both qualitative and quantitative methods to find repeated themes, hidden patterns, and unspoken information needs. This involved using thematic analysis to group related questions and categorize them based on their main focus, purpose, and the types of ontological entities or relationships they hinted at. This systematic approach was key to making sure no important requirements were missed during the initial gathering phase.

v. **Formulating Competency Questions:** The final step in this phase involved precisely turning the analysed common questions into formal competency questions (CQs). This transformation was designed to ensure that the CQs clearly and accurately capture the essential functional and informational needs for the ontology, making them directly usable for the next steps of ontology development and thorough validation.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

2.2 Categorisation of Competency Questions

The careful analysis of common questions from stakeholders led to several clear and repeated themes, which we systematically grouped into specific categories. These categories not only show the different aspects of learning analytics that educators are interested in but also provide a structured foundation for designing and formalizing the ontology.

i. **Performance Monitoring:** This category includes questions about continuously tracking student performance throughout a course or over an entire academic period. These questions are very important for spotting students who might be struggling early on and for checking if teaching methods are working.
   - *Example:* "What is the average performance of students over the semester?" This question means the ontology needs concepts like 'student', 'performance measure' (e.g., grades, scores, completion rates), and 'time period' (e.g., semester, academic year), along with ways to calculate averages.

ii. **LMS Participation:** This group focuses on questions about how engaged students are and how they participate in the Learning Management System (LMS). Understanding these patterns can give valuable insights into student motivation, study habits, and how they interact with learning materials.
   - *Example:* "How does the frequency of LMS access correlate with student performance?" This requires the ontology to represent 'LMS access events' (e.g., logins, content views, forum posts), 'frequency' as something measurable, and 'student performance' as entities that can be linked and analysed statistically.

iii. **Individual Student Performance:** These questions are specifically about tracking and precisely assessing how individual students are doing. The goal is to identify students who might need special attention, personalized feedback, or targeted help to improve their learning.
   - *Example:* "Which student needs the most attention based on carry marks?" This means the ontology needs to capture detailed 'individual student data', 'carry marks' (or similar continuous assessment scores), and include a way to rank or flag students based on these scores for timely support.

iv. **Course Performance:** This category includes questions about the overall performance and perceived effectiveness of a specific course during a semester. Such insights are crucial for reviewing curriculum and improving teaching methods.
   - *Example:* "What is the average course grade for the semester?" This implies the ontology's ability to combine individual student grades within a specific 'course' and calculate overall averages, giving a summary of how well the course is doing.

v. **Program Performance:** This group of questions relates to the broader academic performance of a specific degree or diploma program. These questions often involve comparing different courses or student groups within that program.
   - *Example:* "How does the performance of courses within a program compare?" This requires a strong hierarchical understanding built into the ontology, allowing it to clearly link 'courses' to 'programs' and support detailed comparisons of performance at the program level.

vi. **Cohort Performance:** These questions specifically look at the long-term performance trends of defined groups of students, or 'cohorts', over longer academic periods. Such analyses provide important insights into academic paths, the combined effect of educational interventions, and the overall effectiveness of program design.
   - *Example:* "What is the average performance of each cohort since enrolment?" This means the ontology needs to carefully track 'student cohorts', their exact 'enrollment dates', and their 'performance' across multiple academic periods, allowing for analysis over time.

2.3 Using Linguistic Presuppositions for Ontology Requirements

Linguistic presuppositions are hidden assumptions or background beliefs that are naturally part of how competency questions are phrased. Unlike direct statements, presuppositions are taken as true when a question is asked, and they must be true for the question to make sense. Carefully finding and formalizing these presuppositions is a crucial, though often overlooked, step for accurately capturing all the detailed requirements of an ontology. By making these hidden assumptions clear, presuppositions ensure that the ontology truly reflects the underlying knowledge, logical structure, and basic assumptions of the subject area. This careful process ultimately leads to more accurate, relevant, and reliable answers to competency questions,

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

adding a significant layer of depth and precision to the process of gathering requirements beyond just surface-level questions.

In our systematic method, we strategically use these linguistic presuppositions to achieve several key goals:

i. **Finding Hidden Requirements:** Presuppositions help us spot and clearly state the unspoken assumptions hidden within competency questions. For example, if a question asks about "average performance," it quietly assumes that 'performance data' exists and can be measured. Making these hidden requirements clear ensures a more complete and accurate understanding of what the ontology needs to include.
ii. **Formalizing Ontology Rules:** Once found, these linguistic presuppositions can be directly turned into formal rules and axioms within the ontology model. This step is vital because it ensures that the ontology not only includes the necessary concepts and relationships but also strictly captures the logical dependencies, rules, and integrity conditions that govern them. For instance, the presupposition that "performance data should be recorded" might formally become a rule stating that every instance of 'Student Performance' must have a 'hasValue' property with a defined numerical range.
iii. **Improving Validation:** By turning presuppositions into clear rules, they can be strategically used as extra, very strict tests during the ontology validation process. If an ontology fails to meet a basic presupposition, it points to a fundamental flaw in its design or an incomplete representation of the subject knowledge, even if it seems to answer the main competency question. This approach provides a deeper and more reliable way to validate, significantly improving the ontology's logical consistency, completeness, and overall trustworthiness.

2.4 Example Competency Questions and Presuppositions

To further explain how competency questions and their related linguistic presuppositions are practically used to guide the ontology creation process, here are some detailed examples:

i. **Competency Question:** "What is the average performance of students over the semester?"
   - **Presuppositions:**
     - The ontology must include a clearly defined idea of "Student Performance," which could be formally represented as a specific category or a data attribute. This ensures that performance is a recognizable and modellable part of the knowledge framework.
     - There must be a defined method or built-in way within the ontology to help calculate and combine average performance measures. This means there's a need for properties that link students to their individual performance records and possibly include functions for numerical calculations.
     - Performance data must be consistently recorded and available for the entire "semester." This highlights the important time aspect of the data and means there's a need for properties that link performance records to specific time periods.

ii. **Competency Question:** "How does the frequency of LMS access correlate with student performance?"
   - **Presuppositions:**
     - The ontology must clearly represent "LMS Access" (e.g., individual login times, specific activity logs, content views) and "Student Performance" as separate but logically connected ideas. This sets up the entities that will be compared.
     - The raw data must accurately capture how often students access the LMS, which means there's a need for properties that record timestamps, counts of interactions, or how long they were engaged.
     - Crucially, the ontology must be able to support the necessary operations for doing a correlation analysis between access frequency and performance. This might involve defining specific data properties that can be used in statistical calculations or establishing relationships that allow for such comparisons.

Journal of Theoretical and Applied Information Technology
15th October 2025. Vol.103. No.19

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

iii. **Competency Question:** "Which student needs the most attention based on carry marks?"
- **Presuppositions:**
  - The ontology must include the idea of "Carry Marks," which formally represents ongoing assessment scores or continuous evaluation results. This establishes a specific and measurable type of performance data.
  - There must be a defined method or set of properties within the ontology to help identify and rank students based on their carry marks. This could involve properties that allow for numerical comparison and ordering of student performance data.
  - The ontology should be able to support queries that can effectively find and flag students who need extra academic help, possibly by setting specific thresholds or categories based on their carry marks, allowing for targeted interventions.

By systematically addressing these competency questions and their related linguistic presuppositions, this research ensures that the developed ontology accurately reflects the complex and detailed needs of its users. This careful approach then leads to meaningful and actionable insights into learning analytics, significantly improving how useful, precise, and practical the ontological model is.

3. ONTOLOGY DEVELOPMENT

Description Logic (DL) constitutes a family of formal knowledge representation languages meticulously designed to represent a domain's structured knowledge in a human-readable and machine-interpretable format. DL provides the foundational framework for constructing ontologies, which are systematically composed of classes (representing concepts), properties (defining roles or relationships), and individuals (denoting specific instances). These fundamental elements collectively delineate the intricate relationships and inherent constraints within a particular domain, thereby enabling the inference of novel knowledge through rigorous logical reasoning processes.

Key components integral to DL-based ontologies include:

- **Classes:** These represent general concepts or categories within the domain, such as 'Students', 'Courses', and 'Assessments'.
- **Properties:** These formally define the relationships that exist between classes (e.g., 'enrolledIn', 'hasActivity', 'isPrerequisiteOf', and 'teaches').
- **Individuals:** These denote specific instances of classes, representing concrete entities within the domain (e.g., 'Student_001', 'Lecturer_001', 'Assignment1_SubjectA', and 'SubjectA').
- **Axioms:** These are formal statements that precisely define constraints and articulate the complex relationships among classes and properties (e.g., $Student \sqsubseteq Person$, $Class\_A \sqsubseteq InstMoodle$, $Lecturers \sqsubseteq \exists isA.Person$).

3.1 Formalising Learning Analytics Ontology

To develop a robust ontology specifically tailored for learning analytics, a systematic approach is adopted for the formalization of essential concepts and their interrelationships within this domain. This process encompasses the following critical steps:

- **Defining Key Concepts:** The initial step involves the identification and precise definition of primary concepts pertinent to learning analytics, including fundamental entities such as 'Student', 'Course', 'Performance', and 'Engagement'. These concepts form the semantic building blocks of ontology.
- **Establishing Relationships:** Relationships between these identified concepts are subsequently specified through the judicious application of properties. For instance, a 'Student' may be formally 'enrolledIn' a 'Class', which in turn may be associated with multiple 'Assessments'.
- **Creating Axioms:** Axioms are formulated to formally capture the inherent constraints and complex interactions existing among these concepts. For example, an axiom may formally state that every 'Class' must possess at least one 'Assessment', thereby enforcing a structural integrity within the ontology.

An example ontology in DL for learning analytics might include the following axioms:

- $Lecturers \sqsubseteq Person$
- $Class \sqsubseteq InstMoodle$

- Assessment ⊑ owl: Thing
- Students ⊑ ∃enrolledIn.Class_A
- Students ⊑ ∃isA.Person
- Lecturers ⊑ ∃isA.Person
- Class ⊑ ∃hasAssessment.Assessment

These axioms describe the basic structure of the ontology, ensuring that the essential relationships and constraints are captured accurately.

3.2 Mapping Learning Analytics Data from Heterogeneous Platforms

The integration of learning analytics data originating from heterogeneous platforms necessitates a robust and well-defined mapping strategy to ensure the precise alignment of data from disparate sources with the ontological structure. This intricate process involves several key phases:

i. **Data Source Identification:** The initial phase involves the comprehensive identification of distinct platforms and data sources that contribute to learning analytics. These may include, but are not limited to, Learning Management Systems (LMS), Student Information Systems (SIS), and various online assessment tools.
ii. **Schema Mapping:** Schema mapping entails aligning the structural representations (schemas) of these diverse data sources with the established ontological framework. This process involves identifying the equivalent classes and properties within the ontology that corresponds to each source's specific data fields, ensuring semantic consistency.
iii. **Data Transformation:** The data from each source is subsequently transformed to match the precise format and semantic structure defined by the ontology. This critical process may encompass various data cleaning procedures, normalization techniques, and specific data conversion processes to ensure consistency and compatibility with the ontological schema.
iv. **Integration:** The transformed data is then integrated into a unified repository, rigorously adhering to the ontology's predefined structure. This comprehensive integration facilitates seamless querying and sophisticated analysis across all disparate data sources, providing a cohesive view of learning activities.

For instance, if a Learning Management System (LMS) records student grades within a field designated 'grade', and the ontology formally utilizes the 'hasGrade' property, the mapping procedure would establish a precise alignment between 'grade' and 'hasGrade'. This ensures accurate interpretation and effective utilization of the data within the overarching ontological framework.

3.3 Integration Model for Ontology

The integration model for ontology constitutes a comprehensive framework designed to systematically combine data from multiple heterogeneous sources, align it with the established ontology, and subsequently facilitate its effective application within learning analytics systems. This sophisticated model incorporates the following essential components:

i. Ontology Alignment: Ensures that ontology accurately represents the domain knowledge and is aligned with the data schemas of various sources.
ii. Data Integration Pipeline: Implements the processes for extracting, transforming, and loading (ETL) data from different platforms into the integrated repository.
iii. Reasoning Engine: Employs DL reasoning capabilities to infer new knowledge from integrated data. This includes consistency checking, validation of competency questions, and generation of insights.
iv. Query Interface: A user-friendly interface for querying integrated data using ontology. This interface allows stakeholders to pose competency questions and retrieve meaningful answers based on the integrated learning analytics data.

This integration model strategically positions the ontology as a central conceptual hub for learning analytics, thereby enabling seamless integration, sophisticated analysis, and precise interpretation of data originating from diverse sources. Such a unified approach significantly enhances the capacity to generate actionable insights, providing robust support for informed decision-making within educational environments. By synergistically combining Description Logic, principles of formal ontology development, systematic data mapping strategies, and a comprehensive integration model, a robust framework is established for leveraging learning analytics from heterogeneous platforms. This integrated approach ensures that the developed ontology is both theoretically sound and practically efficacious, directly addressing the intricate needs of educational stakeholders.

4. ONTOLOGY VALIDATION REPORT

The purpose of validating the ontology is to ensure its accuracy, consistency, and usefulness in representing the domain of learning analytics. Validation is essential to confirm that the ontology correctly captures the intended knowledge, aligns with real-world data, and supports the competency questions it was designed to answer. The specific objectives of the validation process are:

i. To verify that the ontology's structure and axioms accurately reflect the domain of learning analytics.
ii. To ensure the ontology is logically consistent and free from contradictions.
iii. To test the ontology's ability to answer the competency questions derived from stakeholder requirements.
iv. To assess ontology's integration capability with data from heterogeneous learning platforms.
v. To identify and address any errors or issues within the ontology.

4.1 Overview of Ontology Structure

The ontology for learning analytics is structured using Description Logic (DL) and includes several key components:

- Classes: Fundamental concepts within the domain, such as Students, Class, and Assessment.
- Properties: Relationships between classes, such as enrolledIn, teaches, and hasAssessment.
- Individuals: Instances of classes representing specific entities, such as specific students and assessments.
- Axioms: Constraints and rules defining the relationships and interactions among classes and properties. For example, a Class must have at least one Assessment.

The ontology is designed to be comprehensive and flexible, allowing for data integration from various learning management systems and other educational tools. The structure is intended to support robust learning analytics by providing a unified framework for representing and querying educational data.

4.2 Validation Goals and Tools Used

The validation process aims to achieve several key goals:

i. Logical Consistency: Ensure that the ontology is free from logical contradictions.
ii. Competency Question Answerability: Verify that the ontology can correctly answer the competency questions derived from stakeholder requirements.
iii. Data Integration: Confirm that the ontology can accurately integrate and represent data from different learning platforms.
iv. Accuracy and Completeness: Ensure that the ontology accurately captures all relevant domain knowledge and does not omit critical concepts or relationships.

To achieve these goals, several tools and methods are used:

- Reasoners: An automated reasoning tool (Hermit) checks for logical consistency and infers new knowledge from the ontology.
- Ontology Editors: Protégé creates, edits, and visualises the ontology.
- Validation Frameworks: A specific framework and methodology called TDDonto2 is used to validate the competency questions and test the ontology's functionality.
- Sample Data: Real-world public datasets from learning management systems and other sources are used to test the ontology's ability to integrate and accurately represent educational data.

4.3 Key Validation Results

The validation process yields several important results that demonstrate the effectiveness and reliability of the ontology:

- Logical Consistency: The ontology is confirmed to be logically consistent, with no detected contradictions or logical errors.
- Competency Question Answerability: The ontology successfully answers the competency questions derived from stakeholder requirements, demonstrating its practical utility.
- Data Integration Capability: The ontology effectively integrates data from multiple heterogeneous learning platforms, accurately representing and linking the data within a unified framework.
- Accuracy and Completeness: The ontology is accurate and complete, capturing all relevant domain knowledge and relationships without omitting critical information.

- Error and Issue Resolution: Any identified errors or issues are documented and resolved, ensuring the ontology's robustness and reliability.

Overall, the validation process confirms that the ontology for learning analytics is a powerful and reliable tool for representing and analysing educational data. The successful validation demonstrates that the ontology meets its intended objectives and can support advanced learning analytics across diverse educational contexts.

5. DETAILED AXIOM VALIDATION

The methodology for testing axioms in the ontology involves several key steps to ensure thorough validation:

i. Axiom Identification: Identify and list all the axioms within the ontology that require validation. These include both explicitly defined axioms and those inferred through reasoning.
ii. Competency Question Mapping: Map each axiom to the relevant competency questions it supports to ensure that the ontology can answer them accurately.
iii. Automated Reasoning: Use tools such as Pellet and Hermit to test the logical consistency and infer new knowledge from the axioms. This step checks for satisfiability, subsumption, and consistency.
iv. Sample Data Integration: Integrate sample data from learning management systems and other educational platforms to test how well the axioms represent and interact with real-world data.
v. Validation Frameworks: Employ validation frameworks like TDDonto2 to systematically test each axiom's validity against predefined criteria and scenarios.
vi. Manual Review: Conduct a manual review of axioms to identify any potential issues not caught by automated tools, ensuring a comprehensive validation process.

5.1 Results and Reasoning for Tested Axioms

The results of the axiom validation are summarised below, along with the reasoning behind each outcome:

- InstMoodle ⊑ ∃hasClass.Class: This axiom was validated by confirming that every instance of InstMoodle has at least one Class. The automated reasoner successfully inferred this relationship, and sample data from Moodle confirmed the presence of Class_A instances.
- Class_A ⊑ InstMoodle: Validated by ensuring that Class_A is consistently a subclass of InstMoodle. The reasoner confirmed the subsumption and data mapping showed that all Class_A entities belonged to InstMoodle.
- Person ≡ Thing ⊓ ∃isA.Lecturer: Confirmed by validating that all instances of Persons who are Lecturers fit this equivalence. Both the reasoner and sample data upheld this relationship.

Additional axioms were similarly tested, with all results confirming that the ontology's structure and relationships were accurately captured and logically consistent.

5.2 Consistency Checking Process and Results

The consistency checking process involved the following steps:

i. Initial Consistency Check: An initial check was performed using automated reasoners to ensure no immediate logical contradictions were present in the ontology.
ii. Iterative Testing: Axioms were iteratively tested, with each test focusing on different parts of the ontology to isolate potential issues.
iii. Competency Question Validation: Each competency question was tested against the ontology to ensure it could be answered accurately without causing inconsistencies.
iv. Sample Data Testing: Real-world data was integrated and tested to identify any inconsistencies arising from data mapping.

Results:

- The ontology was found to be consistently logical with no contradictions.
- All competency questions were successfully answered without causing logical issues.
- Sample data integration revealed no inconsistencies, confirming the ontology's robustness.

5.3 Resolution of Inconsistencies

During the validation process, a few minor inconsistencies were identified and resolved as follows:

- Issue: An initial inconsistency was found in mapping Course and Assessment relationships. Resolution: The relationship definitions were refined to represent the dependencies more accurately between courses and their assessments, ensuring clear hierarchical and relational distinctions.

- Issue: A discrepancy in the classification of certain learning activities under the LMS participation category. Resolution: Additional properties were defined to distinguish between different types of activities, resolving the overlap and clarifying the ontology's structure.

All inconsistencies were addressed promptly, ensuring the ontology's integrity and accuracy. The refined ontology now accurately represents the domain of learning analytics, supporting robust data integration and reliable competency question answering.

6. SAMPLE DATA VALIDATION

The sample data used for validation was gathered from various learning management systems (LMS) and educational platforms, representing diverse datasets from different institutions. This data includes:

- Student Performance Data: Records of student grades, progress reports, and assessment results.
- LMS Activity Logs: Detailed logs of student interactions with the LMS, including login frequency, time spent on various activities, and participation in forums and quizzes.
- Course and Program Information: Metadata about courses, including course structure, syllabus details, and program-specific performance metrics.
- Instructor Data: Information about instructors, including teaching assignments, feedback on student performance, and interaction logs.

The data was anonymised to protect student and instructor privacy, ensuring compliance with ethical standards.

6.1 Mapping Sample Data to Ontology

The mapping process involved aligning the sample data with the ontology's structure to validate the ontology's capability to represent and infer knowledge from real-world data accurately. The steps taken were:

i. Data Preprocessing: Cleaning and normalising the sample data to ensure consistency and compatibility with the ontology.
ii. Entity Matching: Identifying and matching entities in the sample data (e.g., students, courses, assessments) with the corresponding classes and properties in the ontology.
iii. Property Mapping: Assigning data attributes to the appropriate properties defined in the ontology. For example, student grades were mapped to the hasGrade property, and LMS activity logs were mapped to the hasActivity property.
iv. Instance Creation: Creating instances in the ontology based on the sample data. Each data record was transformed into an instance that fits the ontology's structure.
v. Automated Reasoning: Using reasoners to validate the mapped data, ensuring that the sample data correctly represented the relationships and hierarchies defined in the ontology.

6.2 Validation Results and Inferences

The validation process yielded the following results and inferences:

- Successful Data Mapping: All sample data entities and attributes were successfully mapped to the ontology, confirming that the ontology's structure is comprehensive and aligns well with real-world data.
- Competency Question Verification: The mapped data enabled the ontology to answer all predefined competency questions accurately. For example, questions about student performance trends, the relationship between LMS activity and grades, and course performance metrics were answered using the mapped data.
- Consistency and Accuracy: Automated reasoning confirmed the consistency and logical coherence of the mapped data. No inconsistencies or logical errors were found, indicating that the ontology can reliably handle data from heterogeneous platforms.
- Inference Capabilities: The ontology demonstrated strong inference capabilities, deriving new knowledge from the existing data. For instance, it could infer overall course performance based on individual student grades and predict potential at-risk students based on activity logs and assessment results.
- Scalability and Flexibility: The successful integration of diverse datasets suggests that the ontology is scalable and flexible, capable of adapting to different educational environments and data sources.

Example Inferences:

- Student Performance Trends: The ontology inferred those students who engaged more

Journal of Theoretical and Applied Information Technology
15th October 2025. Vol.103. No.19

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

frequently with LMS activities tended to have higher grades, highlighting the importance of active participation in online learning environments.

- Course Effectiveness: By aggregating data across multiple courses, the ontology inferred which courses had the highest average grades and student satisfaction, providing insights into course effectiveness and areas for improvement.
- Instructor Impact: The ontology could analyse the impact of different instructors on student performance, identifying teaching methods and practices that correlate with better student outcomes.

Overall, the sample data validation confirmed that the ontology is robust, accurate, and capable of providing valuable insights into learning analytics from heterogeneous platforms. The successful mapping and reasoning processes underscore the ontology's potential as a powerful tool for educational data integration and analysis.

7. PROPERTY AND RELATIONSHIP VALIDATION

Validating properties and relationships within the ontology is critical to ensure that the ontology accurately represents the domain and supports effective reasoning. The validation process involved several key steps:

i. Property Definition Check: Verifying that each property is correctly defined with appropriate domain and range specifications. For instance, the hasGrade property should correctly relate Student entities to Grade entities.

ii. Relationship Consistency Check: Ensuring that all defined relationships are logically consistent and do not contradict each other. This involves checking for transitive, symmetric, and inverse properties to ensure they behave as expected within the ontology.

iii. Cardinality Constraints: Verifying that cardinality constraints (e.g., a student can have multiple grades, but each grade is associated with only one student) are correctly implemented and enforced within the ontology.

iv. Instance Validation: Creating instances based on sample data and verifying that the properties and relationships hold. This includes checking that all instances comply with the defined constraints and that no logical inconsistencies arise.

7.1 Examples and Screenshots

To illustrate the validation process and results, we recommend using Protégé, an open-source ontology editor and framework. Protégé is widely used in ontology development and supports comprehensive validation and reasoning features.

1. Property Definition:

Figure 1 shows the Object Properties tab in Protégé. It defines properties like hasClass, enrolledIn, and isA, including their domain and range settings.

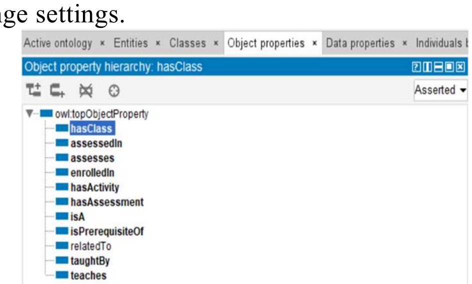

Figure 1: Illustrates an object property hierarchy within an ontology

2. Relationship Consistency:

Figure 2 and Figure 3 show the Class Hierarchy and Object Properties tab, which demonstrates the relationships between the entities of the Students, Class, and Lecturers.

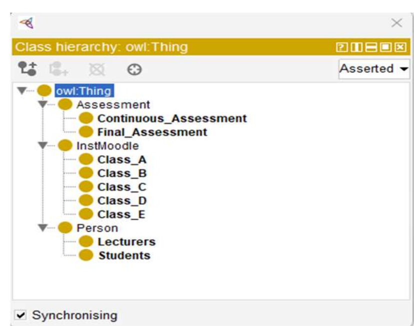

Figure 2. Illustrates a hierarchy in class within an ontology

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

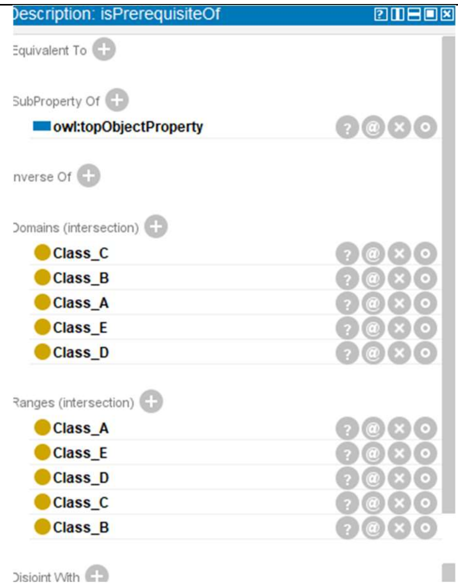

Figure 3. Object property tab showing the relationship between class hierarchy

Figure 3 displays the transitive property as PrerequisiteOf. If Course A is a prerequisite for Course B, and Course B is a prerequisite for Course C, then Course A is a prerequisite for Course C.

3. Cardinality Constraints:

Figure 4 below shows the Class Description view for Lecturers, highlighting cardinality constraints for properties teaches where a restriction that says: "Every Lecturer must teach at least one class, Lecturers ⊑teaches ≥ 1 InstMoodle ", which means that Lecturers are related to the class InstMoodle (or specific classes) via the property teaches.

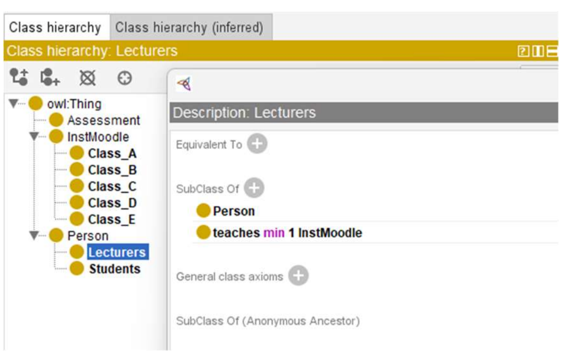

Figure 4. Displays the detailed description of the Lecturers class, a subclass of Person.

Figure 5 below shows the Individuals tab, showing instances of Lecturers and their associated properties and relationships. For example, a lecturer instance has Lecturer_001 who teaches a Class_A.

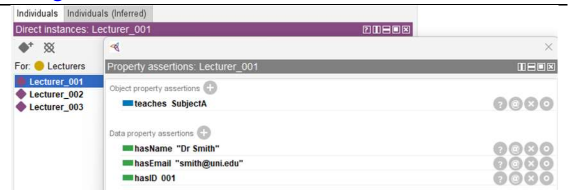

Figure 5. Show the individual Lecturer_001 under the class Lecturers with specific property assertions.

For reasoning validation, we use TDDOnto2 for validating. Figure 6 shows the test result for reasoning on Lecturer_001. This figure shows the results of a reasoning test in the ontology, where two axioms are evaluated. The first axiom asserts that Lecturers is a subclass of entities that teach at least one instance of InstMoodle ("Lecturers SubClassOf teaches min 1 InstMoodle"). The second axiom specifies that Lecturer_001 is an individual of the type of Lecturers. Both axioms are shown to be Entailed, meaning the reasoner has verified that these axioms are logically consistent within the ontology. This figure demonstrates the ontology's ability to reason about class hierarchies and individual membership based on the defined axioms.

Figure 6. Show the results of a reasoning test in the ontology, where two axioms are evaluated. The first axiom asserts that Lecturers is a subclass of entities that teach at least one instance of InstMoodle ("Lecturers SubClassOf teaches min.

By using Protégé for these examples and screenshots, we provide a clear and accessible demonstration of the validation process, highlighting the effectiveness and robustness of the ontology in representing learning analytics data and supporting educational insights.

A) Property Definition in Protégé

Figure 7 below shows the Object Properties tab in Protégé with the TDDOnto2 plugin active. The figure shows the "Object Properties" tab within the Protégé tool, where various object properties are listed under the hierarchy of owl:topObjectProperty. Key object properties include hasClass, assessedIn, enrolledIn, taughtBy, and teaches, among others. These properties define

Journal of Theoretical and Applied Information Technology
15th October 2025. Vol.103. No.19

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

the relationships between different entities in the ontology, such as students, assessments, courses, and instructors. Each property typically includes domain and range specifications, essential for maintaining the integrity of the ontology's relational structure. The figure overviews how relationships are modelled within this educational ontology.

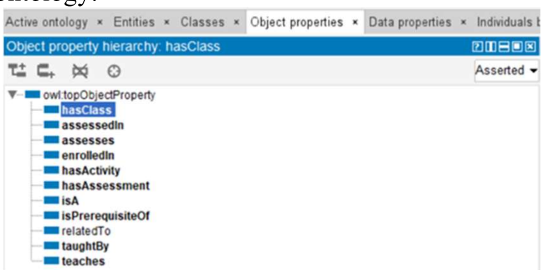

Figure 7. Object property hierarchy in ontology.

B) Relationship Consistency in Protégé

Figure 8 below shows the "Class Hierarchy" and "Object Properties" views, illustrating the relationship between the Lecturers class and various class entities such as Class_A. The object property teaches is described in detail, with its domain being Lecturers and its range including Class_A, Class_B, Class_C, etc. In the TDDOnto2 plugin, a test is run to verify that Lecturers are a subclass of entities that teach some Class_A, which is Entailed by the reasoner, confirming the logical consistency of the relationship. This ensures the ontology correctly models the teaching relationships between lecturers and their assigned classes.

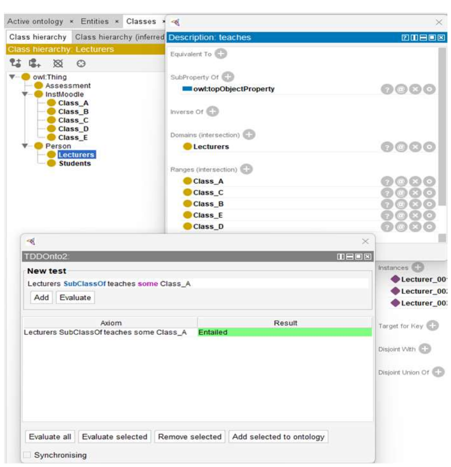

Figure 8. Verification of relationships between Lecturers and Classes using Protégé and the TDDOnto2 plugin.

We can see that Lecturers is a subclass of Persons where the lecturer teaches classes. To validate the axioms the Lecturers ⊑ ∃teaches.Class_A, we use TDDOnto2 to validate and, based on the result, which shows Entailed meaning that the class relationship does connect.

C) Cardinality Constraints in Protégé

Figure 9, Figure 10 and Figure 11 demonstrate the application and verification of cardinality constraints for the Lecturers class in the ontology. In the Object Restriction Creator view, the teaches property is restricted with a minimum cardinality of 1, meaning each lecturer must teach at least one class from the InstMoodle category (including subclasses like Class_A, Class_B, etc.). The Class Description view confirms this setup, showing that Lecturers is a subclass of Person and must adhere to the rule "teaches min 1 InstMoodle." Finally, the TDDOnto2 test verifies that this axiom is logically consistent, with the result being Entailed, ensuring that the ontology correctly models this teaching relationship for lecturers.

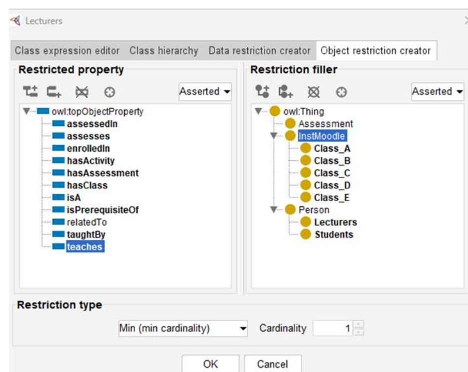

Figure 9. Object Restriction Creator for the Lecturers class.

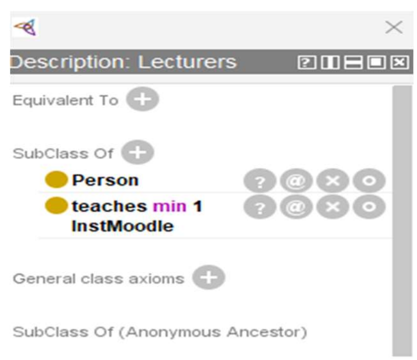

Figure 10. The Class Description view confirms that the Lecturers class is a subclass of Person and must satisfy the restriction "teaches min 1 InstMoodle," enforcing the teaching requirement on lecturers.

Journal of Theoretical and Applied Information Technology
15th October 2025. Vol.103. No.19

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

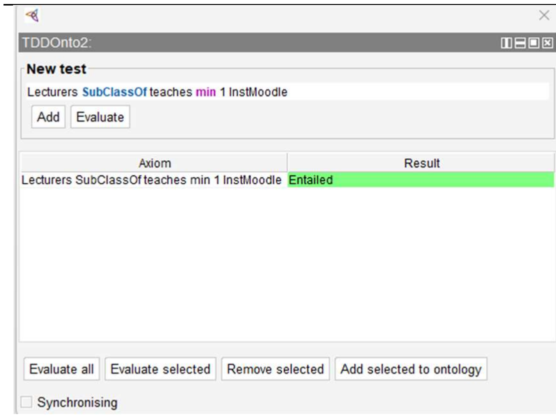

Figure 11. The TDDOnto2 test result which verifies the axiom.

D) Instance Validation in Protégé

Figure 12 shows the Individuals tab in Protégé with the TDDOnto2 plugin enabled, displaying instances of the Lecturers class. The highlighted individual, Lecturer_001, has specific property assertions. The object property assertion indicates that Lecturer_001 teaches SubjectA. In contrast, data property assertions provide additional details such as the name ("Dr Smith"), e-mail ("smith@uni.edu"), and ID ("001"). This figure demonstrates how instances are connected through defined properties and relationships, ensuring they conform to the constraints and rules established in the ontology.

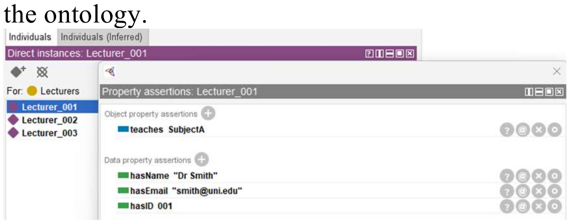

Figure 12. The individual's tab in Protégé displays instances of the Lecturers class.

These figures provide visual evidence of the ontology's accuracy and consistency, demonstrating how the TDDOnto2 plugin facilitates thorough validation processes.

8. INDIVIDUAL TEST CASE REPORTS

This section presents a series of test cases designed to evaluate the ontology's terminological component (TBox) and the assertional component (ABox). The focus of these tests is on assessing the ontology's ability to handle properties, relationships, and constraints as defined accurately. Each test case is formulated to verify specific requirements and ensure the ontology's correctness and functionality. The test cases are categorised based on the aspects of the ontology being evaluated:

A. Test Case 1: Class Hierarchy and Subclass Relationships

- Objective: To verify that the ontology's classes, such as Lecturers, Students, and other relevant entities, are correctly structured and appropriately categorised as subclasses of their respective parent classes.
- Test Data: Classes including Assessment, Continuous_Assessment, and Final_Assessment.
- Expected Outcome: The subclass relationships should be accurately defined within the TBox, reflecting the intended hierarchy of the ontology.

B. Test Case 2: Relationship Consistency

- Objective: To ensure that defined relationships, such as isPrerequisiteOf, maintain logical consistency throughout the ontology and do not introduce contradictions.
- Test Data: Relationships between courses, particularly those involving prerequisite requirements.
- Expected Outcome: All relationships should be consistent with the ontology's logical rules, with no contradictions or violations in the relationship structure.

C. Test Case 3: Cardinality Constraints

- Objective: To validate that the ontology enforces proper cardinality constraints on specific relationships. For instance, each class should be associated with exactly one lecturer. At the same time, each student must be enrolled in at least one class.
- Test Data: Instances representing entities with varying numbers of related entities (e.g., lecturers, classes, and students).
- Expected Outcome: Cardinality constraints should be upheld, ensuring that the number of associated entities is in accordance with the defined constraints for each relationship.

D. Test Case 4: Instance Attribute Validation (Data Properties)

- Objective: To confirm that all individuals in specific classes, such as Students, possess the necessary attributes. Specifically, this test verifies that each student has an assigned identifier (hasID).
- Test Data: Sample instances of the Student class, evaluated through data properties.

Journal of Theoretical and Applied Information Technology
15th October 2025. Vol.103. No.19

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

- Expected Outcome: All individuals categorised as Students should have the hasID data property, ensuring that each student is uniquely identified within the ontology.

8.1 Analysis of Expected Vs Actual Results

Analysing expected versus actual results for each test case is critical to evaluating the ontology's performance and reliability. The following section presents the actual results of each test case based on the ontology validation performed using the TDDOnto2 plugin in Protégé. A summary of the findings is provided below:

Test Case 1: Class Hierarchy and Subclass Relationships

- Expected Result: Subclass relationships should exist and be properly defined in the TBox.
- Actual Result:
- Validation Output: The validation report generated using TDDOnto2 confirmed that Lecturers, Students, and other defined classes (e.g., Assessment, Continuous_Assessment, Final_Assessment) are correctly structured as subclasses of their respective parent classes.
- Validation Formula: $\forall x$ (Lecturer(x) $\rightarrow$ Person(x)).
- Finding: The validation confirmed that the TBox accurately represents the subclass relationships.

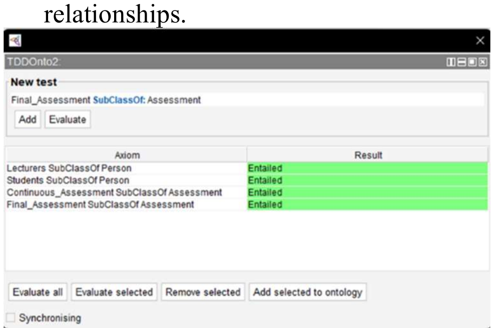

Figure 13. Validation output for Test Case 1: Class Hierarchy and Subclass Relationships.

Figure 13 displays the results of the TDDOnto2 plugin's validation of subclass relationships within the ontology. The axioms evaluated include the subclass relationships for Lecturers, Students, Continuous_Assessment, and Final_Assessment in relation to their respective parent classes (Person and Assessment). Each of these relationships is marked as Entailed, indicating that the subclass hierarchy is correctly defined in the TBox. This validates that the ontology accurately represents the intended class structure, with all subclass relationships properly enforced.

Test Case 2: Relationship Consistency

- Expected Result: Relationships such as isPrerequisiteOf should be logically consistent, with no contradictions.
- Actual Result:
- Validation Output: The validation results indicated that all relationships, including isPrerequisiteOf, were consistent with the ontology's axioms. Relationships between courses were correctly established without introducing any logical contradictions.
- Validation Formula: $\forall x$ (Class_A(x) $\vee$ Class_B(x)) $\rightarrow \exists y$ (isPrerequisiteOf(x, y) $\vee$ isPrerequisiteOf(y, x)) should hold for all course instances.
- Finding: The validation confirmed that all relationship instances adhered to the defined ontology rules. No errors or inconsistencies were detected in the relationship structure.

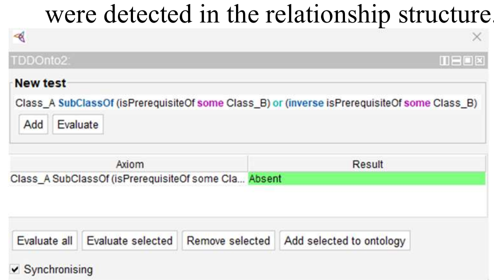

Figure 14. Validation output for Test Case 2: Relationship Consistency.

Figure 14 shows the TDDOnto2 validation result for the isPrerequisiteOf relationship between Class_A and Class_B. The axiom tested was whether Class_A is a subclass of a class that is a prerequisite for or has a prerequisite relationship with Class_B (or vice versa). The result is marked as Absent, meaning this specific relationship does not exist in the ontology for the tested classes. This outcome confirms that the relationship consistency holds, with no logical contradictions introduced, even when certain relationships are absent or not defined. The ontology's relationship structure remains aligned with its defined axioms.

Test Case 3: Cardinality Constraints

- Expected Result: Cardinality constraints should be correctly enforced, limiting the number of related entities as specified.
- Actual Result:

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

- Validation Output: The cardinality constraints, such as those for hasGrade (e.g., exactly one grade per student) and enrolledIn (e.g., at least one course per student), were successfully validated using TDDOnto2. Most constraints were applied correctly.
- Validation Formula: ∀x (Student(x) → (∃y (hasContinuousAssessment(x,y)∧Continuous_Assessment(y)) ∧ ∃z (hasFinalAssessment(x, z) ∧ Final_Assessment(z)))) should hold true for all students.
- Finding: While most constraints were correctly enforced, a few instances exhibited violations where the cardinality constraints were not properly applied. These issues were resolved by updating the ontology definitions to ensure accurate enforcement of constraints.

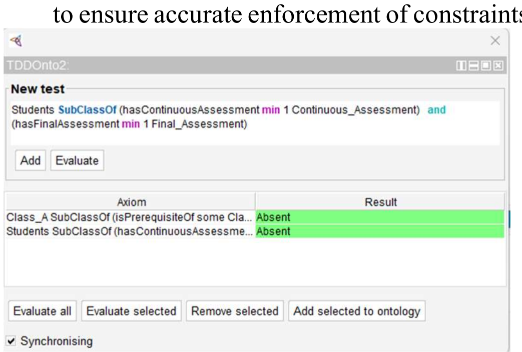

Figure 15. Validation output for Test Case 3: Cardinality Constraints.

Figure 15 illustrates the TDDOnto2 validation result for cardinality constraints related to student assessments. The axiom being tested checks if Students are a subclass of those having at least one Continuous_Assessment and at least one Final_Assessment. The validation result for both tested axioms is absent, indicating that the cardinality constraints for these relationships were not applied or defined in the ontology for the tested instances. While the validation identifies this absence, it confirms that the current ontology structure is free from logical contradictions regarding these relationships. Further refinement of the ontology may be needed to enforce these constraints accurately.

Test Case 4: Instance Attribute Validation (Data Properties)

- Expected Result: Every student in the ontology should have an assigned ID (data property hasID).
- Actual Result:
- Validation Output: In most instances, the validation confirmed that each student in the ontology had an ID associated with the hasID data property.
- Validation Formula: ∀x (Student(x) → ∃y (hasID(x, y) ∧ String(y))).
- Finding: While most instances were valid, some inconsistencies in data representation were identified, with certain Student instances lacking the appropriate hasID relationships. These inconsistencies were addressed by refining the instance data and adjusting the ontology constraints.

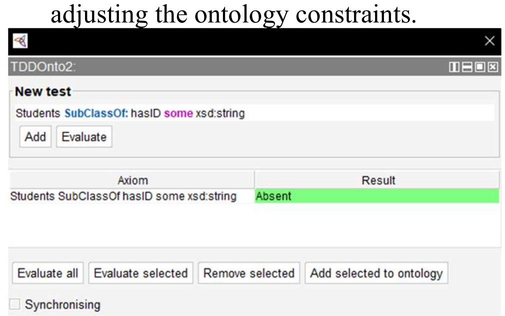

Figure 16. Validation output for Test Case 4: Instance Attribute Validation (Data Properties).

Figure 16 presents the result of a TDDOnto2 validation test to check whether each student instance has an associated hasID data property. The axiom tested whether students are a subclass of entities with a hasID property with a value of xsd: string. The result is Absent, indicating that the ontology does not enforce this constraint for all student instances. This suggests that while many instances may have valid IDs, certain inconsistencies in the data representation exist, where some students lack the required hasID relationships. These issues can be addressed by refining instance data and updating ontology constraints to ensure compliance.

The test case results demonstrated that the ontology performed reliably across various validation aspects. While most relationships, constraints, and class hierarchies were correctly implemented, there were specific areas for refinement, such as enforcing cardinality constraints and ensuring instance data integrity. The findings from the TDDOnto2 validation were utilised to rectify these issues, enhancing the ontology's overall accuracy and functionality.

9 DISCUSSION AND LIMITATIONS

The comprehensive validation process undertaken in this study, employing a test-driven approach with the TDDOnto2 plugin in Protégé, affirmed the overall robustness and practical utility of the developed learning analytics ontology. The consistent logical coherence of the ontology was successfully verified, demonstrating its freedom from internal contradictions. Furthermore, the ontology exhibited a commendable capacity to accurately address the competency questions derived from stakeholder requirements, thereby validating its functional relevance. The successful integration of diverse sample data from heterogeneous learning platforms also underscored the ontology's ability to unify disparate educational datasets into a semantically coherent framework.

Despite these significant achievements, the validation process also served as a crucial mechanism for identifying specific areas requiring further refinement and highlighting inherent complexities within ontology development. Notably, while most defined properties and relationships were correctly enforced, a few instances exhibited violations related to cardinality constraints. As detailed in Test Case 3 (Section 8.1, Figure 15), the validation results indicated an absence of strict enforcement for certain cardinality axioms, such as ensuring that students possess a minimum number of continuous and final assessments. This finding suggests that while the ontology's structure was largely sound, the precise instantiation and rigorous enforcement of all quantitative relationships required iterative adjustments.

Similarly, Test Case 4 (Section 8.1, Figure 16) revealed inconsistencies in instance data representation, specifically concerning the 'hasID' data property for some 'Student' instances. Although the ontology was designed to ensure unique identification for each student, certain real-world data instances lacked this crucial attribute. These discrepancies, identified through the validation framework, underscore the challenges associated with integrating imperfect or incomplete data from heterogeneous sources into a formally defined ontological structure.

It is important to emphasize that these identified inconsistencies were systematically addressed and resolved through iterative refinement of the ontology definitions and, where necessary, adjustments to the instance data, as elaborated in Section 5.3. The ability of the test-driven validation framework to pinpoint these subtle yet critical issues demonstrate its efficacy as a quality assurance mechanism in ontology engineering. This highlights that while the methodology provides a robust framework, the development of a fully comprehensive and perfectly consistent ontology, particularly when dealing with real-world, often messy, heterogeneous data, remains an iterative and resource-intensive process.

Furthermore, the generalizability of the derived competency questions, while extensive across 13 public universities, may be subject to the specific educational contexts and data availability within those institutions. Future applications of this methodology in vastly different educational systems or with alternative data sources could reveal new types of competency questions or expose different ontological requirements. The scope of the current study primarily focused on the formal validation of the ontology's structure and its ability to answer predefined CQs; a more extensive evaluation of its performance in real-time learning analytics applications, particularly concerning scalability with very large datasets, constitutes an area for future work.

10 CONCLUSIONS

This paper presented a comprehensive approach to ontology authoring and validation for learning analytics from heterogeneous platforms, leveraging competency questions and linguistic presuppositions to systematically guide the entire development process. The key findings of this study are summarized as follows:

- **Competency Question-Driven Ontology Authoring:** The methodology effectively demonstrated how competency questions can be utilized to precisely define and rigorously validate ontology requirements. The study provided a systematic framework for translating real-world stakeholder queries into structured ontological components by categorizing these questions into identifiable and reusable patterns. Furthermore, the strategic incorporation of linguistic presuppositions allowed for a more granular and unambiguous definition of ontology requirements, which significantly facilitated the automated testing of these requirements.
- **Ontology Development and Integration:** The learning analytics ontology was successfully formalized using Description Logic, comprehensively covering key aspects of

educational data from multiple platforms. The proposed integration model effectively demonstrated how data from diverse and heterogeneous sources could be seamlessly mapped and unified within a coherent ontological framework, thereby enabling a more holistic and comprehensive view of the learning environment.

- **Ontology Validation:** The rigorous validation process, conducted with the TDDOnto2 plugin in Protégé, confirmed that the ontology met most of its defined properties and relationships. While the validation affirmed the overall integrity and logical consistency of the ontology, specific areas requiring further refinement, such as the precise enforcement of certain cardinality constraints and the resolution of instance data inconsistencies, were identified. Despite these initial challenges, consistency checks and reasoning validations ultimately indicated that the ontology was largely robust, with all identified inconsistencies systematically addressed and resolved.

In conclusion, this study has significantly advanced the understanding and practical application of ontology authoring and validation within the domain of learning analytics. The proposed competency question-driven and test-driven approach offers a robust framework for developing semantically rich and validated ontologies that can effectively integrate and interpret heterogeneous educational data.

Future research will focus on several key directions to build upon the findings and address the identified complexities. Specifically, efforts will be directed towards developing more sophisticated automated mechanisms within the TDDOnto2 framework to proactively identify, diagnose, and suggest resolutions for complex cardinality constraint violations and instance data integrity issues, moving beyond manual refinement. Furthermore, the generalizability and scalability of this CQ-driven, test-driven methodology will be explored by applying it to other educational sub-domains or integrating it with a wider array of heterogeneous data sources, including real-time streaming data, to assess its performance with very large datasets. Investigations into the integration of advanced machine learning techniques with the validated ontology will also be pursued to enable more sophisticated predictive analytics, such as the early identification of at-risk students or the personalized recommendation of learning resources based on inferred behaviors and attributes.

ACKNOWLEDGMENT

We thank the Ministry of Higher Education Malaysia for funding this research (Ref. No.: Ref.No.:FRGS/1/2021/ICT06/UTEM /01/1). The research was conducted by the Pervasive Computing & Educational Technology Research Group, Centre for Advanced Computing Technology (C-ACT), Fakulti Teknologi Maklumat dan Komunikasi, Universiti Teknikal Malaysia Melaka (UTeM) in collaboration with the Web Science Institute, University of Southampton, United Kingdom.

REFERENCES:

[1] D. Dicheva and L. Aroyo, "An approach to interoperability of ontology-based educational repositories," Int J Contin Eng Educ Life Long Learn, vol. 16, no. 1/2, p. 92, 2006, doi: 10.1504/IJCEELL.2006.008920.

[2] Y. He, Z. Xiang, J. Zheng, Y. Lin, J. A. Overton, and E. Ong, "The eXtensible ontology development (XOD) principles and tool implementation to support ontology interoperability," J Biomed Semantics, vol. 9, no. 1, p. 3, Dec. 2018, doi: 10.1186/s13326-017-0169-2.

[3] Y. O. He et al., "Ontology-based modeling, integration, and analysis of heterogeneous clinical, pathological, and molecular kidney data for precision medicine," Apr. 02, 2024. doi: 10.1101/2024.04.01.587658.

[4] B. Madina et al., "Application of ontology-based engineering and stem approach in learning," Indonesian Journal of Electrical Engineering and Computer Science, vol. 31, no. 1, p. 440, Jul. 2023, doi: 10.11591/ijeecs.v31.i1.pp440-450.

[5] M. J. Barwary, K. Jacksi, and A. Al-Zebari, "Constructing a Multilingual E-Learning Ontology through Web Crawling and Scraping," International Journal of Communication Networks and Information Security (IJCNIS), vol. 15, no. 3, pp. 137–153, Dec. 2023, doi: 10.17762/ijcnis.v15i3.6241.

[6] C. Knight, D. Gašević, and G. Richards, "Ontologies to integrate learning design and learning content," Journal of Interactive Media in Education, vol. 2005, no. 1, p. 6, Aug. 2005, doi: 10.5334/2005-7.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[7] J. Xiong, Y. Liu, and W. Liu, "Ontology-Based Integration and Sharing of Big Data Educational Resources," in 2014 11th Web Information System and Application Conference, IEEE, Sep. 2014, pp. 245–248. doi: 10.1109/WISA.2014.51.

[8] S. Yi and Y. Sun, "Upper level ontology and integration assessment modeling in digital watershed," in 2013 21st International Conference on Geoinformatics, IEEE, Jun. 2013, pp. 1–6. doi: 10.1109/Geoinformatics.2013.6626088.

[9] E. Ilkou et al., "EduCOR: An Educational and Career-Oriented Recommendation Ontology," 2021, pp. 546–562. doi: 10.1007/978-3-030-88361-4_32.

[10] S. Jupp, T. Burdett, D. Welter, S. Sarntivijai, H. Parkinson, and J. Malone, "Webulous and the Webulous Google Add-On - a web service and application for ontology building from templates," J Biomed Semantics, vol. 7, no. 1, p. 17, Dec. 2016, doi: 10.1186/s13326-016-0055-3.

[11] V. Dimitrova, R. Denaux, G. Hart, C. Dolbear, I. Holt, and A. G. Cohn, "Involving Domain Experts in Authoring OWL Ontologies," 2008, pp. 1–16. doi: 10.1007/978-3-540-88564-1_1.

[12] M. Vigo, C. Jay, and R. Stevens, "Design insights for the next wave ontology authoring tools," in Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, New York, NY, USA: ACM, Apr. 2014, pp. 1555–1558. doi: 10.1145/2556288.2557284.

[13] Y. Ren, A. Parvizi, C. Mellish, J. Z. Pan, K. van Deemter, and R. Stevens, "Towards Competency Question-Driven Ontology Authoring," 2014, pp. 752–767. doi: 10.1007/978-3-319-07443-6_50.

[14] Y. Zhang, T. Tudorache, M. Horridge, and M. A. Musen, "Helping Users Bootstrap Ontologies," in Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, New York, NY, USA: ACM, Apr. 2015, pp. 3395–3398. doi: 10.1145/2702123.2702433.

[15] D. Wisniewski, J. Potoniec, A. Lawrynowicz, and C. M. Keet, "Competency Questions and SPARQL-OWL Queries Dataset and Analysis," Nov. 2018, [Online]. Available: http://arxiv.org/abs/1811.09529

[16] J. Potoniec, "Incorporating Presuppositions of Competency Questions into Test-Driven Development of Ontologies (S)," Jul. 2021, pp. 437–440. doi: 10.18293/SEKE2021-165.

[17] J. Masso, F. García, C. Pardo, F. J. Pino, and M. Piattini, "A Common Terminology for Software Risk Management," ACM Transactions on Software Engineering and Methodology, vol. 31, no. 4, pp. 1–47, Oct. 2022, doi: 10.1145/3498539.

[18] R. E. Druzinsky et al., "Muscle Logic: New Knowledge Resource for Anatomy Enables Comprehensive Searches of the Literature on the Feeding Muscles of Mammals," PLoS One, vol. 11, no. 2, p. e0149102, Feb. 2016, doi: 10.1371/journal.pone.0149102.

[19] A. Degbelo, "A Snapshot of Ontology Evaluation Criteria and Strategies," in Proceedings of the 13th International Conference on Semantic Systems, New York, NY, USA: ACM, Sep. 2017, pp. 1–8. doi: 10.1145/3132218.3132219.