

Article

A Semi-Automated Ontology Framework for Multi-Level Competency Mapping

Aomsap Inkong-ngarm ¹, Jakramate Bootkrajang ², Samerkae Somhom ² and Areerat Trongratsameethong ^{2,*}

¹ Data Science Consortium, Faculty of Engineering, Chiang Mai University, Chiang Mai 50200, Thailand; aomsap_i@cmu.ac.th

² Department of Computer Science, Faculty of Science, Chiang Mai University, Chiang Mai 50200, Thailand; jakramate.b@cmu.ac.th (J.B.); samerkae.s@cmu.ac.th (S.S.)

* Correspondence: areerat.t@cmu.ac.th

Abstract

Aligning academic transcripts with occupational competency requirements remains challenging because course labels and job-skill terms are semantically ambiguous, role-specific, and difficult to explain. This paper proposes the Ontology Framework for Multi-level Competency Mapping (O4CM), a semi-automated framework integrating a Large Language Model (LLM) ensemble, Human-in-the-Loop (HITL) verification, Sentence-BERT (SBERT) semantic representation, the Path Consistency Index (PCI), and Total Accumulated Competency Score/Normalised Total Accumulated Competency Score (TACS/NTACS) ranking. O4CM was evaluated on a historical job-posting corpus and anonymised transcripts from five university programmes through ablation, sensitivity analysis, baseline comparison, and expert-labelled validation. The LLM ensemble reached high consensus for 21 of 22 Occupational Information Network (O*NET) knowledge-domain mappings (95.45%), each of which was subsequently expert-verified. In a computing-only expert-aligned analysis, the full framework most closely matched expert rankings across three data-domain roles. Within this dataset, ontology-path evidence can support more transparent competency ranking for educational advising and exploratory recruitment screening.

Keywords: ontology-based competency mapping; job type-conditioned ranking; ontology induction; Path Consistency Index; human-in-the-loop verification

1. Introduction

The alignment between university education and labour-market requirements has become a pressing concern in knowledge-intensive industries. In data-domain occupations such as Data Scientist, Data Analyst, and Data Engineer, employers increasingly require specific technical and analytical competencies rather than broad academic attainment alone [1]. This challenge is becoming more acute as artificial intelligence reshapes technically intensive occupations and may contribute to more selective early-career hiring patterns [2]. Consequently, student-to-job matching requires methods that can determine whether transcript-level learning evidence corresponds to the specific competency requirements of a target role rather than relying only on broad indicators such as degree title, programme label, or cumulative GPA.

Researchers have sought to move beyond the grade point average (GPA) metric by framing graduate selection as a person–job fit problem in which semantic representations of job requirements are aligned with representations of candidate qualifications [1]. Early



Academic Editor: Feiping Nie

Received: 10 May 2026

Revised: 18 June 2026

Accepted: 22 June 2026

Published: 30 June 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

methods relied on keyword overlap and bag-of-words similarity, which are computationally inexpensive. However, these methods operate purely at the surface of text and cannot resolve semantic equivalences when academic course names and job-posting skill terms belong to different terminological conventions. The development of dense sentence-embedding models—notably, Sentence-BERT [3]—substantially improved matching quality by encoding meaning rather than vocabulary. Nonetheless, high cosine similarity between a course and a knowledge node does not guarantee structural coherence within a knowledge hierarchy, nor does it reflect how strongly that node is demanded by the specific occupation under consideration. Ontological knowledge representation offers a principled way to address this limitation: by organising domain knowledge into explicitly defined hierarchical structures, ontologies allow a course to be evaluated not only against the node it most resembles but also against the broader semantic context defined by the skill group and job category in which that node is embedded [4]. Constructing such ontologies manually, however, is labour-intensive and difficult to sustain as labour-market requirements evolve. Although Large Language Models (LLMs) can support semi-automated ontology construction, fully automated generation remains risky because Natural Language Generation (NLG) models may produce fluent but unfaithful or unverifiable outputs; such risks can be further affected by decoding and inference choices that may increase the likelihood of hallucinated content [5]. Therefore, semi-automated frameworks that combine LLM scalability with the semantic rigour of human-expert oversight represent a promising and increasingly active research direction.

Against this background, the present study identifies three inter-related technical problems that are frequently addressed in isolation but are rarely integrated within a single end-to-end competency-matching pipeline. The first is semantic surface ambiguity: A course may achieve high similarity relative to a knowledge node due to incidental vocabulary overlap while remaining misaligned with the underlying competency that the node represents in practice. For instance, courses such as Human Relations, Calculus 1, and Lanna Studies can exhibit strong leaf-level similarity relative to computing-related knowledge nodes, even though they do not develop computing competency, which can propagate false positives when hierarchical context is not verified. The second problem is role indiscrimination: Conventional matching approaches often treat data-domain jobs as a single category and therefore struggle to differentiate candidates across closely related roles such as Data Scientist, Data Analyst, and Data Engineer, despite meaningful differences in the competencies emphasised by these positions. The third problem is a lack of transparency: Numerical ranking scores provide limited insight into which courses and competency areas drive the final decision. Because black-box decision-support systems can be difficult for users and stakeholders to understand, insufficiently explained rankings may raise concerns about accountability and fairness in high-stakes settings such as academic advising and graduate recruitment [6,7].

These issues motivate a framework that combines ontology-guided structure, multi-level verification to suppress surface-level drift, job-type-conditioned scoring, and traceable evidence-based explanations.

Prior work has made important advances but leaves these problems collectively unaddressed. Studies on skill extraction from job postings [8] and on résumé-to-job matching [1] have demonstrated the feasibility of large-scale semantic alignment using dense representations, but these approaches operate with flat, non-hierarchical skill structures and remain vulnerable to surface ambiguity. Ontology-based competency modelling frameworks [4] have shown that structured knowledge representations improve matching reliability, yet they typically depend on manually curated ontologies that are costly to maintain. The reliability of LLM-induced ontology structures under systematic ensemble validation and

HITL oversight has not been empirically characterised, and the contributions of individual framework components to overall performance have rarely been assessed through rigorous ablation analysis. Evaluation methodologies in the literature are also inconsistent: many studies report aggregate similarity or ranking correlation metrics without determining whether a framework correctly discriminates among semantically adjacent job roles or produces outputs that practitioners can act upon [9]. Furthermore, the use of transcript-based academic evidence as structured input to ontology-grounded competency matching has not been studied in Southeast Asian higher education contexts, where curriculum diversity across engineering, business, science, and arts faculties provides an initial cross-domain test bed for discrimination between computing-oriented and non-computing curricula, while more difficult near-boundary curricula remain to be evaluated in future work.

To address these gaps, this study proposes a semi-automated, ontology-grounded framework for student-to-job competency matching. Practically, the proposed framework provides a principled, data-driven mechanism for aligning student records with job-market requirements. Methodologically, the Path Consistency Index (PCI) and Total Accumulated Competency Score (TACS) mechanisms extend existing embedding-based approaches by incorporating multi-level ontological verification and job-side structural relevance signals, yielding more discriminative rankings than the flat-SBERT baseline evaluated in this study. With respect to explainability, black-box ranking systems can provide limited insight into how particular recommendations are produced, which is a central concern in explainable AI [7].

The framework is also designed to be extensible across domains: the Occupational Information Network (O*NET) knowledge taxonomy [10], the LLM ensemble protocol, and the TACS and NTACS scoring formulations may be adapted to other academic systems and occupational categories, provided that an appropriate seed taxonomy, domain-specific job-posting corpus, and expert validation process are available.

In summary, we consider the following to be our main contributions:

1. We propose a semi-automated Ontology framework for multi-level Competency Mapping (O4CM), a semi-automated ontology-grounded framework that integrates LLM-assisted ontology induction; expert validation; multi-level structural verification via the PCI; and job-conditioned competency scoring operationalised by the raw TACS and its normalised ranking score, the Normalised Total Accumulated Competency Score (NTACS).
2. We evaluate the framework's structural consistency, semantic representation design, and discriminative performance through ontology validation; semantic representation diagnostics; PCI-based group separation; systematic ablation analysis; and comparison with four reference methods spanning GPA ranking, keyword matching, flat Sentence Bidirectional Encoder Representations from Transformers (flat SBERT) matching, and knowledge-node-only ontology matching.
3. We demonstrate the framework's capacity to produce interpretable ranking outputs through semantic traceability maps, which decompose each candidate's competency score into course-level and skill-group-level evidence contributions.

A positioning of O4CM relative to related approaches, together with a detailed discussion of the three gaps that motivate the present work, is provided in Section 2.4.

Several limitations of the present study should be noted. First, the empirical evaluation is based on a single Thai university and a static job-posting corpus, limiting generalisation claims. Second, the current framework addresses three data-domain roles and one contrasting arts domain; broader occupational coverage remains to be explored in future work. Third, despite HITL verification, LLM-generated augmentation can still introduce semantic drift [11], and the quality of the induced ontology depends on the availability of

suitable seed taxonomies and qualified domain experts. Fourth, the framework currently uses a structural relevance definition for $IF_{c,k}$ rather than raw demand frequency; hybrid formulations have not yet been explored.

The remainder of this article is organised as follows. Section 2 reviews related work on skill extraction from job postings and identifies the specific gaps that motivate the present framework. Section 4 presents the proposed five-phase semi-automated ontology framework for competency mapping. Section 5 reports the empirical evaluation. Section 6 interprets the findings in relation to prior literature and discusses practical implications for educational counselling and talent acquisition. Section 7 summarises the main contributions, acknowledges limitations, and outlines directions for future research.

2. Related Work

2.1. Competency-Based Student-to-Job Matching and Skill Extraction

Candidate-to-job matching has been approached through several theoretical frameworks, including collaborative filtering and bilateral recommendation theory [12], automated profile ranking using multi-criteria scoring [13], and ontology-based semantic skill extraction [1]. Collaborative filtering approaches model matching as a preference-prediction problem but require historical rating data and cannot account for the structural content of academic credentials. Keyword-based and profile-ranking systems support structured candidate evaluation yet rely on surface-form overlap that cannot resolve semantic equivalence between course evidence and occupational knowledge requirements [13]. GPA-based screening [12] ignores individual course content entirely and cannot differentiate candidates by occupational relevance. Ontology-based competency frameworks define the mapping between demonstrated knowledge and target-occupation requirements as a multi-level alignment problem where structured competency evidence discriminates qualified applicants more reliably than unstructured CV parsing [13] and ontology-anchored representations substantially improve skill-to-taxonomy alignment over surface-form matching [1]. A recent systematic review of job recommender systems covering 76 studies confirms that ontology and semantic web approaches represent a well-established methodology in this domain [14].

Skill extraction grounds competency modelling in Information Extraction (IE) and labour-market analytics [15]. Named Entity Recognition (NER) transforms unstructured job advertisements into structured skill and qualification entities, but three persistent challenges limit reliability: vocabulary mismatch, polysemy, and granularity variation [16,17]. Standardised taxonomies such as O*NET [18] and ESCO [19] address these through synonym resolution and hierarchical classification, providing reusable knowledge layers that enable consistent cross-source skill normalisation.

2.2. Ontology- and Embedding-Based Competency Representation

Ontology-based competency frameworks offer formal, machine-interpretable representations of knowledge and skills that support interoperability and systematic competency assessment [20]. However, many existing models rely on manual ontology engineering, which is costly and difficult to scale [20]. Standardised taxonomies mitigate this by providing reusable semantic anchors, with O*NET [18] and ESCO [19] being the most widely adopted in recruitment and educational contexts.

Semantic similarity methods have evolved from sparse lexical representations such as TF-IDF to dense static word embeddings and, more recently, contextual language representations. Word2Vec introduced efficient architectures for learning continuous word vectors capturing syntactic and semantic regularities from large-scale corpora [21]. BERT advanced language representation learning through deep bidirectional Transformer en-

coders that produce context-dependent representations [22], but standard BERT is not optimised for large-scale pairwise similarity search because sentence pairs must be jointly encoded for comparison. Sentence-BERT (SBERT) addresses this through a Siamese bi-encoder architecture that generates fixed-size sentence embeddings that are comparable via cosine similarity [3], enabling scalable similarity without pairwise overhead [23]. However, a flat SBERT approach applied directly to student course records and job-skill atoms has three limitations: it ignores hierarchical ontology relationships, omits job-type-specific relevance signals, and degrades on short course or skill labels that lack sufficient semantic context. These limitations motivate hierarchical path verification as a mechanism that is complementary to embedding-based matching.

2.3. LLM-Assisted Ontology Construction, HITL Verification, and Explainable Ranking

Large language models (LLMs) support knowledge graph construction and ontology induction through triple extraction, relation inference, and schema alignment [24]. Their application to competency framework mapping has been demonstrated by Jemal et al. [25], who show that LLM-based approaches can align heterogeneous competency frameworks by exploiting the semantic generalisation capabilities of large language models. However, LLM use in ontology construction is limited by stochastic outputs, hallucination risks [5], and weak global structural consistency. Constrained prompt templates address these limitations by enforcing machine-readable outputs and definition-grounded mapping decisions, and ensemble agreement across multiple models serves as a preliminary consensus signal that reduces model-specific variation [26]. However, because high cross-model agreement does not necessarily imply semantic correctness [5], ensemble consensus is not sufficient as a standalone acceptance criterion.

HITL verification is widely used in knowledge engineering to ensure that machine-generated structures are semantically appropriate, contextually grounded, and suitable for downstream use [5]. In LLM-assisted ontology construction, expert review is particularly important because generated mappings may be fluent and internally consistent while still being misaligned with domain-specific meaning or professional practice. Dual expert perspectives—from industry professionals with competency-assessment experience and academic experts in curriculum design—are especially valuable for preserving both professional relevance and pedagogical validity.

Explainable AI (XAI) is important in AI-assisted recruitment and educational assessment because automated matching systems must provide transparent and accountable justifications for their recommendations [6]. Model-level explanation methods such as LIME and SHAP [27,28] estimate feature contributions in predictive models but do not produce traceable semantic links among educational records, skills, knowledge domains, and job requirements. In recruitment contexts, heterogeneous resume formats and non-standard document structures complicate information extraction, automatic matching, and candidate ranking [29], while in educational contexts, heterogeneous curricular documents and course descriptions make interpretable course-skill-job alignment difficult [30]. Therefore, ontology-based frameworks that surface explicit multi-level links between course content, intermediate skill groups, and job-relevant knowledge domains offer a principled path towards structural explainability that goes beyond feature-level attribution.

2.4. Positioning of the Present Study

The reviewed literature reveals three gaps that collectively motivate the present work. At the structural level, to the best of our knowledge, no prior framework integrates multi-level hierarchical path verification with LLM ensemble induction under formal majority voting and HITL escalation that includes expert review across all agreement tiers, includ-

ing unanimous outputs. At the scoring level, existing approaches rarely unify job-type-conditioned structural relevance weighting, transcript evidence, and auditable competency decomposition within a single ranking pipeline. At the evaluation level, component-level ablation and cross-domain discrimination beyond a single contrasting pair remain under-explored, particularly in Southeast Asian higher education contexts.

The proposed O4CM framework addresses these gaps through three contributions. First, it constructs a three-level ontology backbone (L_1 : job categories; L_2 : skill groups; L_3 : O*NET knowledge domains) via a semi-automated process in which an LLM ensemble proposes candidate mappings and a dual-panel HITL gate provides the final acceptance decision, replacing both fully manual engineering and unchecked LLM automation. Second, it introduces the Path Consistency Index (PCI) as a continuous structural alignment score that verifies each course record's locked path from L_3 through L_2 to L_1 and combines it with a job-type-conditioned Importance Factor (IF) to produce a normalised competency score (NTACS) that reflects both academic quality and job-side structural relevance simultaneously. Third, it provides a course-level semantic traceability map that links each ranking decision back to specific ontology paths and job-posting evidence, going beyond feature-level attribution to structural explainability. Compared with flat embedding approaches [11], which apply SBERT cosine similarity directly to O*NET labels without hierarchical verification or job-type conditioning, O4CM's multi-level path structure captures curriculum–occupation alignment at multiple semantic granularities.

Table 1 summarises how O4CM differs from representative prior approaches along four key dimensions.

Table 1. Comparison of O4CM with representative prior approaches across four framework dimensions. ✓ = fully present; △ = partial or implicit; × = absent.

Approach	Hierarchical Path Verification	HITL Quality Gate	Job-Conditioned Scoring	Traceability
Flat SBERT matching [11]	×	×	×	×
Manual ontology [4]	✓	×	×	△
LLM-only ontology construction [25]	△	×	×	×
O4CM (proposed)	✓	✓	✓	✓

3. Datasets

This section describes the two datasets used in the experimental evaluation: job-posting corpora for characterisation of labour-market demand and anonymised student transcripts for representation of academic competency evidence.

3.1. Job-Posting Corpora

Two sets of job postings were used: a primary data-domain corpus for competency mapping and evaluation and a small Visual Arts set serving as a negative control.

3.1.1. Data-Domain Corpus

The primary corpus consists of 5715 postings across three data-domain roles: Data Scientist, Data Analyst, and Data Engineer. The postings originate from Indeed.com and were obtained as a publicly released dataset on Kaggle [31], reflecting the United States labour market circa 2018. Because this Indeed-sourced Kaggle dataset is historical and platform-specific, the derived $IF_{c,k}$ values should be interpreted as corpus-conditioned structural relevance signals rather than indicators of current or general labour-market demand. The corpus also predates recent AI- and LLM-era job requirements. Some core technical terms for these roles, such as python,SQL, and machine learning, remain recognisable in contemporary data-domain job descriptions; however, newer AI- and LLM-related vocabulary, including generative AI, prompt engineering, vector databases,

and recent MLOps practices, is not covered by this historical corpus. Accordingly, when current labour-market alignment is required, the O4CM pipeline should be re-run on a more recent and domain-specific job-posting corpus, with the resulting ontology mappings and $IF_{c,k}$ values subject to corpus-specific validation.

The original creator did not publish a data card, so their search strategy and deduplication procedure cannot be verified; this lack of documentation also means that platform-specific and source-selection bias in the original collection process cannot be ruled out. At the O4CM processing stage, the Phase 1 pipeline performs its own deduplication before downstream mapping (Algorithm 1). The dataset attributes recorded for each posting are detailed in Table 2.

Algorithm 1: Preparation of Job-Skill and Course Atoms (Phase 1)

Input: Job postings $\mathcal{J}$; student transcripts $\mathcal{T}$

Output: Job-skill atoms $\mathcal{A}_{\text{job}}$; course atoms $\mathcal{A}_{\text{stud}}$

Initialise $\mathcal{A}_{\text{job}} \leftarrow \emptyset$, $\mathcal{A}_{\text{stud}} \leftarrow \emptyset$;

foreach posting $j \in \mathcal{J}$ **do**

 Parse skills field $\rightarrow$ *skill_list*;

 Remove stop words, duplicates, special characters;

foreach skill $s \in \textit{skill_list}$ **do**

$\mathcal{A}_{\text{job}} \leftarrow \mathcal{A}_{\text{job}} \cup \{(\text{Job_ID}_j, \text{Job_Type}_j, s)\}$;

foreach transcript $t \in \mathcal{T}$ **do**

foreach course record $r \in t$ **do**

$\mathcal{A}_{\text{stud}} \leftarrow \mathcal{A}_{\text{stud}} \cup \{(\text{Stu_ID}, \text{Course}, \text{Credits}, \text{Grade})\}$;

Remove duplicate entries from $\mathcal{A}_{\text{job}}$ and $\mathcal{A}_{\text{stud}}$;

return $\mathcal{A}_{\text{job}}$, $\mathcal{A}_{\text{stud}}$;

Table 2. Attributes of the job-posting dataset.

Attribute	Description
Job_ID	Unique identifier assigned to each job posting.
Job_Title	Title of the advertised position.
Link	URL reference to the original job posting.
Queried_Salary	Salary information provided in the posting, where available.
Job_Type	Occupational category used for grouping, such as Data Scientist, Data Analyst, or Data Engineer.
Skill	Raw skill field extracted from the posting; subsequently decomposed into atomic skill units during preprocessing.
Company	Name of the organisation that published the posting.
Description	Full textual description of the role, including responsibilities and requirements.

The posting counts and skill density across the three target categories are summarised in Table 3.

Seniority was not explicitly annotated as a separate field in this Indeed-sourced Kaggle dataset; postings were grouped into the three target job types using the existing Job_Type label, with each posting contributing equally, regardless of seniority level. Table 4 details the ten most frequent job titles and skill items in the corpus. Title-level inspection shows that senior, principal, and lead variants account for a measurable share of postings, with Data Scientist (715), Data Analyst (405), and Data Engineer (391) being the most frequent baseline

designations. The skill distribution highlights a strong concentration on programming and data-platform competencies, which serve as the job-side evidence for ontology mapping and structural relevance estimation.

Table 3. Job-posting dataset statistics by job-type category.

Job Type	Total Postings	Mean Skills/Job	Max Skills/Job	Min Skills/Job
Data Scientist	2543	8.49	20	1
Data Analyst	1793	4.49	20	1
Data Engineer	1379	10.84	20	1
Visual Arts (contrast)	10	8.10	16	5
Total	5725			

Table 4. Ten most frequent job titles and skill items in the posting corpus. Job titles indicate the presence of specialist and senior-role variants within the target categories.

Job Title	Count	Skill Item	Count
Data Scientist	715	Python	3325
Data Analyst	405	SQL	3104
Data Engineer	391	Machine Learning	2297
Senior Data Scientist	205	R	2234
Senior Data Engineer	136	Hadoop	1714
Senior Data Analyst	86	Spark	1531
Big Data Engineer	80	Java	1480
Principal Data Scientist	62	Tableau	1236
Lead Data Scientist	49	Data Mining	1059
Sr. Data Scientist	45	Hive	966

3.1.2. Visual Arts Negative-Control Corpus

A set of ten Visual Arts postings was collected directly by the research team from artjobs.com and th.jobsdb.com during the preparation of this study, covering Asia and global listings. All postings are in English, consistent with the encoding assumptions of the SBERT model and LLM ensemble used in Phases 1 and 2. This self-collected Visual Arts corpus serves as a small negative-control sample for assessing whether the framework assigns lower structural relevance to an occupational domain that is intentionally unrelated to the data-domain roles. Together, the two corpora yield 5725 postings and 44,679 job–skill atoms after Phase 1 preprocessing.

3.2. Student Transcript Dataset

The student transcript dataset comprises undergraduate records from a university in Thailand, capturing the academic history of 430 students. In total, the dataset contains 50,040 individual course enrolment records. All records were fully anonymised prior to analysis to comply with ethical research standards. This dataset serves as the empirical basis for evaluating the student-to-job matching framework, providing computing-oriented profiles, together with an intentionally contrasting arts-oriented profile for evaluation.

To ensure consistent semantic mapping and performance evaluation, each enrolment record captures specific academic and demographic details. These fields include bilingual course identifiers, credit weights, and achieved grades, which collectively form the raw inputs for the subsequent competency scoring. The complete schema of the extracted attributes is detailed in Table 5.

Table 5. Attributes of the student transcript dataset.

Attribute	Description
std_id_pri	Anonymised unique identifier assigned to each student.
credit	The credit unit weighting of the enrolled subject.
grade_en	The letter grade achieved by the student in the subject.
id_sub	Unique subject code or identifier.
subject_name	The official name of the subject in Thai.
subject_name_en	The translated name of the subject in English.
major_name	The academic programme or major of the student.
faculty_name	The faculty under which the student is enrolled.

The students in this dataset are drawn from five distinct academic programmes. Four of these are computing-related disciplines (Computer Engineering, Business Information Systems, Computer Science, and Information Technology), forming the positive group expected to demonstrate strong alignment with data-domain job requirements. In contrast, the Visual Arts programme is included as a negative control group. This arts-focused curriculum provides a negative-control contrast for the framework, examining its ability to assign low structural similarity relative to academically dissimilar profiles when semantic and hierarchical evidence is weak. Table 6 outlines the curricular orientation and group assignment of each programme.

Table 6. Academic programme descriptions and group assignments.

Programme	Group	Curricular Orientation
B.Eng. Computer Engineering	Positive	Engineering-faculty programme emphasising hardware systems, software development, and applied engineering principles; broadest curriculum in the sample with the widest elective range.
B.B.A. Business Information Systems	Positive	Business-faculty programme with substantial computing content (39.5% of 147 subjects), including data-domain courses such as Python Programming, Big Data Analytics, Data Mining, and Artificial Intelligence; classified in the positive group on the basis of its computing and information-systems orientation.
B.Sc. Computer Science	Positive	Science-faculty programme with a structured theoretical and applied computing foundation; highly standardised curriculum with minimal elective variation.
B.Sc. Information Technology	Positive	Practice-oriented programme emphasising applied systems and IT management; compact and focused subject inventory with low within-programme variability.
B.F.A. Visual Arts	Control	Arts-faculty programme centred on studio practice, design theory, and creative arts disciplines; minimal overlap with computing and data-domain knowledge; included as a negative control to examine whether the framework assigns lower structural similarity to academically dissimilar profiles.

Beyond the curricular differences, the dataset exhibits variations in academic performance and enrolment volume across the groups, which influenced the methodological design. Table 7 summarises the student counts, mean enrolled courses, and GPA statistics per programme. Two observations are particularly relevant. First, the B.F.A. Visual Arts group has the highest mean GPA in this dataset. A GPA-only evaluation would therefore tend to favour this group, despite its weaker alignment with data-domain competency requirements. This highlights the need for a job-conditioned competency score rather than a purely grade-based ranking. Second, programmes differ in the number of courses taken per student. This variation in transcript evidence volume motivates the use of a normalised job-conditioned competency score to support fairer comparisons across candidates. The formal ranking score is defined in Section 4.7.3.

Table 7. Transcript dataset statistics by academic programme. GPA is reported on a 4.0 scale. The four computing programmes form the positive group ($N = 311$); Visual Arts forms the negative control group ($N = 119$).

Programme	Number of Students	Mean of Enrolled Courses per Student	Mean GPA
B.Eng. Computer Engineering	146	52.79	2.74
B.B.A. Business Information Systems	71	48.06	2.81
B.Sc. Computer Science	33	45.48	2.93
B.Sc. Information Technology	61	43.62	2.92
B.F.A. Visual Arts (control)	119	48.45	3.24

In the full transcript dataset, the four computing programmes define the positive group, and Visual Arts defines the control group. The empirical evaluation reported in Section 5 uses subsets drawn from these groups; the subset composition and selection criteria are specified in Section 5.1.

4. Methodology

This section describes the methodology of the proposed semi-automated framework for multi-level semantic knowledge extraction and competency mapping. The framework integrates LLM-assisted induction, ontology engineering, HITL verification, semantic representation, and structural scoring to support traceable student-to-job matching.

4.1. Key Concepts and Notation

Four core concepts underpin this framework.

Ontology: A formal knowledge representation that defines concepts, properties, and hierarchical relationships, supporting logical inference and cross-class relations [32]. Unlike a taxonomy (simple parent–child hierarchy), an ontology enables auditable path-based reasoning. Here, knowledge domains, skill groups, and job categories form a three-level `rdfs:subClassOf` hierarchy.

Large Language Model (LLM): A neural model trained on large text corpora to generate structured natural language outputs [5]. An ensemble of five LLMs proposes candidate ontology mappings; all outputs are subject to HITL verification before materialisation.

Sentence-BERT (SBERT): A Siamese bi-encoder that produces fixed-length sentence embeddings for efficient cosine-similarity computation [3]. The `all-MiniLM-L6-v2` variant embeds both ontology node descriptions and transcript course records.

Human in the Loop (HITL): A design pattern in which human experts review and correct machine-generated outputs at critical decision points [5]. HITL verification is the final acceptance gate for every LLM-proposed mapping before ontology materialisation, including unanimous-agreement cases.

4.2. Framework Overview

The proposed framework, denoted as the Ontology Framework for Multi-level Competency Mapping (O4CM), is organised as a five-phase pipeline. Figure 1 presents the workflow and illustrates how the output of each phase is validated and propagated to the subsequent stage.

The semi-automated design pairs LLM-driven concept induction, which is scalable but hallucination-prone [5], with HITL verification, which is intended to improve semantic appropriateness and auditability.

As illustrated in Figure 1, the five sequential phases are: atomic unit extraction (Phase 1), LLM-assisted ontology construction (Phase 2), semantic augmentation and SBERT encoding (Phase 3), PCI-based structural verification (Phase 4), and job-aware competency scoring (Phase 5).

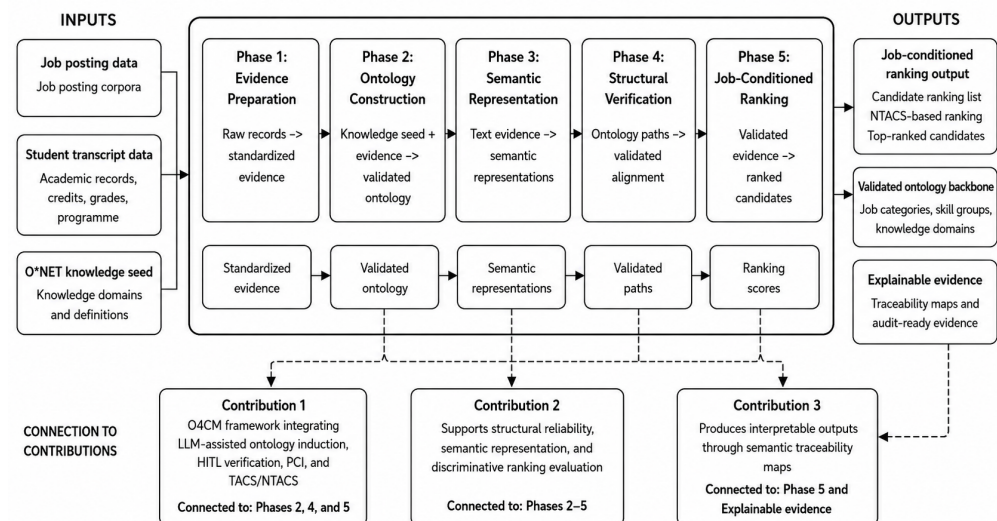


Figure 1. Overview of the O4CM framework. Solid arrows indicate the primary flow of data and processing across the five phases, whereas dashed arrows indicate the conceptual connections between framework components, outputs, and the stated research contributions.

4.3. Phase 1: Atomic Unit Extraction and Preprocessing

Phase 1 converts the heterogeneous inputs described in Section 3—namely, job postings and student transcript records—into standardised atom-level records for ontology-based mapping. The preprocessing procedure consists of three main operations: duplicate records are removed to avoid inflated frequency counts, non-informative special characters and formatting artefacts are removed to reduce textual noise, and credit values in the transcript records are converted into numeric form to support weighted competency scoring in later phases.

Table 2 lists the job-posting attributes. Skill fields stored as string representations of arrays are parsed and expanded into skill atoms as displayed in Figure 2, with each atom occupying a single row for consistent counting and mapping.

Atomic Unit Extraction: Before and After
Before (raw data): - Job_ID: J1029 - Job_Type: Data Scientist - Skill: ['Python', 'SQL', 'Machine Learning', 'TensorFlow'] After (atomic unit extraction): The posting is expanded into individual skill atoms; each record contains exactly one skill: (J1029, Data Scientist, Python) (J1029, Data Scientist, SQL) (J1029, Data Scientist, Machine Learning) (J1029, Data Scientist, TensorFlow) Rationale: Unrolling ensures that each skill becomes a single independently comparable unit, enabling consistent frequency counting, semantic similarity computation, and ontology mapping in subsequent phases.

Figure 2. Illustrative example of atomic unit extraction applied to a single job-posting record. The raw Skill field is parsed and expanded so that each extracted skill atom occupies one row for downstream similarity computation and ontology mapping.

4.4. Phase 2: Semi-Automated Bottom-Up Ontology Construction

Phase 2 constructs the ontology backbone that enables multi-level semantic mapping between job requirements and student learning evidence. Building on the atomic and standardised units produced in Phase 1, this phase follows a bottom-up strategy that begins with well-defined knowledge anchors and progressively induces higher-level groupings and relations. The objective is to reduce manual ontology engineering effort while preserving semantic clarity, structural consistency, and traceability.

In this study, ontology construction is anchored to the O*NET knowledge taxonomy [10], which provides standardised knowledge definitions suitable for occupational competency analysis. The scope is restricted to 22 O*NET knowledge domains that correspond to the selected occupational scope and support the computer-art contrast used in the evaluation. These domains form the seed layer (L_3) and serve as fixed semantic anchors for the induction of skill-group structures (L_2) and the assignment of these groups to job-level categories (L_1). The output of this phase is a validated three-level Web Ontology Language class hierarchy linking knowledge domains, skill groups, and job categories through explicit subclass assertions.

4.4.1. Ontology Construction with LLM Assistance

Ontology construction is implemented as a semi-automated induction process. LLMs are used to propose candidate skill-group structures and mapping assertions under strict output constraints. These outputs are not treated as final ontology assertions. Instead, ensemble agreement is used as a preliminary consensus signal, while HITL verification serves as the final acceptance gate before any mapping is materialised into the ontology.

The workflow consists of three primary tasks:

- **Skill-Group Induction:** The 22 O*NET knowledge domains and their definitions are provided to the LLMs to induce candidate skill groups and concise descriptions. These groups provide the intermediate layer (L_2) between knowledge domains (L_3) and job categories (L_1) but are not treated as final ontology commitments until their associated mappings pass HITL review.
- **Knowledge-to-Skill Mapping ($L_3 \rightarrow L_2$):** Each O*NET knowledge domain is assigned to one relevant induced skill group through definition-grounded classification. This task is the central semantic decision point because it links the fixed O*NET knowledge anchors to the induced skill-group structure and therefore receives the main ensemble-agreement and expert-validation analysis.
- **Skill-to-Job Category Assignment ($L_2 \rightarrow L_1$):** Each validated skill group is assigned to exactly one job category (:Computer_Job or :Art_Job) using a forced-choice semantic dominance criterion. The decision uses the skill-group definition, its assigned knowledge-domain members, and the definitions of the two target job categories.

For ontology materialisation, entities across all three layers are modelled as classes rather than individual instances. Hierarchical subsumption is therefore represented using `rdfs:subClassOf` rather than `rdf:type`. This modelling choice treats the hierarchy as an analytical competency taxonomy rather than a realist claim that knowledge domains are literally skills. It also supports clear semantic separation and enables PCI-based auditing of the locked path from knowledge domains to skill groups and job categories.

Therefore, the resulting mappings should be interpreted as controlled analytical assignments designed for traceable competency mapping, not as complete representations of all possible relationships among knowledge domains, skills, and job categories.

4.4.2. Prompt Design and Constraints

Because prompt design directly influences the induced ontology structure, prompts are engineered to reduce ambiguity and enforce structured outputs suitable for ontology materialisation. To control output variability, the prompts enforce single-label classification and require decisions to be grounded in the provided definitions rather than keyword matching. For the $L_2 \rightarrow L_1$ assignment, extension grounding is applied by providing the LLMs with the knowledge domains already assigned to each skill group as concrete semantic evidence for classification. Outputs must conform to a machine-readable template. For example, the $L_3 \rightarrow L_2$ mapping is constrained to the following RDF-style triple format:

(Knowledge_Name, :belongs_to_skill, Skill_Group_Name)

The relation expressed as :belongs_to_skill is used only as an intermediate parsing label in the LLM output. After HITL verification, accepted mappings are converted into `rdfs:subClassOf` axioms for ontology materialisation and downstream path auditing.

This enables deterministic parsing and conversion into ontology axioms with minimal manual reformatting. The objectives, inputs, constraints, and output schemas for the three induction tasks are described above.

4.4.3. LLM Ensemble and Majority Voting

To minimise model-specific bias and stochastic variation, Phase 2 employs an ensemble of five LLMs: GPT-5.4, GPT-5.3, Gemini 3 Pro, Claude Opus 4.6, and Claude Sonnet 4.6. Each model receives the identical prompt templates described in Section 4.4.2 and is executed independently for every mapping decision.

Majority voting is used to estimate preliminary cross-model consensus and to prioritise cases for expert review. Final acceptance, however, requires HITL verification for all mappings. This voting step serves as a consensus signal rather than an automatic acceptance mechanism, helping to identify low-consensus or malformed outputs that require closer inspection. This design is motivated by the known risk of model-specific errors and hallucinated outputs in LLM-generated content [5].

Each preliminary majority outcome is classified into one of three agreement tiers reflecting the degree of cross-model consensus:

- Unanimous (5/5): All five models agree, indicating high observed cross-model agreement.
- High Majority (4/5): Four models agree, with one dissenting label recorded for inspection.
- Simple Majority (3/5): Three models agree, indicating a weaker consensus signal; all such cases are automatically escalated for mandatory HITL review.

HITL review is also triggered when two or more models produce malformed outputs and no majority can be determined. Under HITL review, a domain expert examines all five candidate labels against the relevant O*NET definitions and assigns a final expert-confirmed label with a documented justification. Only mappings that have passed HITL verification, informed by the ensemble agreement results, are converted into OWL/RDF subclass assertions. The quantitative outcomes of this protocol are reported in Section 5.3.

For clarity and reproducibility, Table 8 summarises the full ensemble configuration, the majority voting rule, and the HITL escalation criteria, together with the exception-handling logic for malformed outputs.

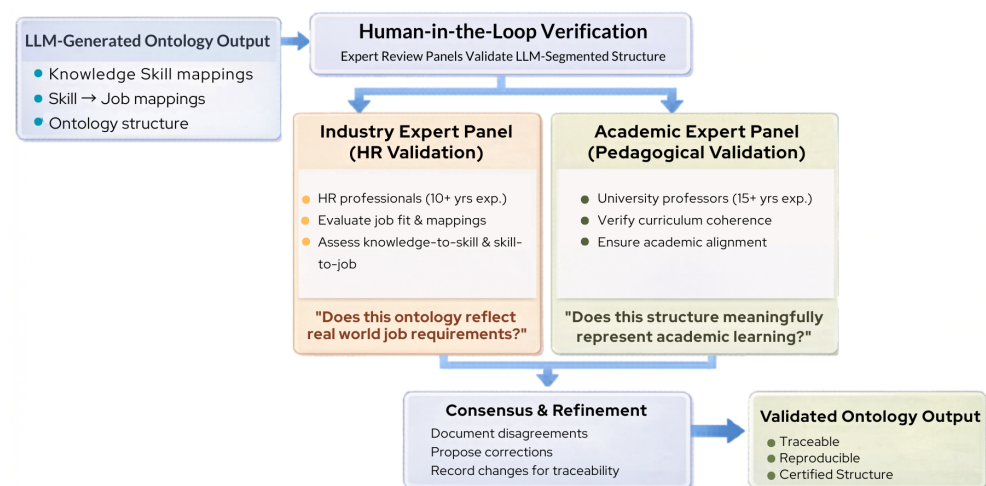
Table 8. LLM ensemble and majority-voting protocol for Phase 2. All models use the structured prompt templates described in Section 4.4.2. Empirical outcomes are reported in Section 5.3.

Component	Configuration	Decision Rule	Exception Handling
Model ensemble	Five LLMs: GPT-5.4, GPT-5.3, Gemini 3 Pro, Claude Opus 4.6, and Claude Sonnet 4.6. Each model is prompted independently with identical inputs.	No model is assigned a privileged weight; all five votes are treated equally.	If a model returns a system error, the response is excluded, provided ≥ 3 valid responses remain; otherwise, the case is escalated to HITL review or rerun using the same prompt template.
Majority voting	Applied to the two classification tasks: knowledge-to-skill mapping and skill-to-job assignment.	Record the preliminary majority label with $\geq 3/5$ agreement and classify the outcome as Unanimous (5/5), High Majority (4/5), or Simple Majority (3/5).	If ≥ 2 outputs are malformed and no majority can be reached, escalate to HITL review.
HITL escalation	Triggered for detailed review by: (i) any simple majority (3/5) outcome or (ii) voting failure from ≥ 2 malformed outputs.	A domain expert reviews all candidate labels against O*NET definitions and assigns the final expert-confirmed label with a documented justification.	All HITL decisions are logged to support reproducibility auditing and prompt refinement.

4.4.4. Human-in-the-Loop Verification Protocol

Because LLM-induced structures are treated as candidate drafts rather than final ontology assertions, HITL verification is applied at the end of Phase 2. The verification confirms that the ensemble-generated mappings (Section 4.4.3) are semantically appropriate for the target domain and structurally consistent for downstream ontology-based reasoning.

Critically, unanimous consensus (5/5 agreement) among the five LLMs is not treated as a sufficient condition for accepting a label without expert inspection. Therefore, HITL verification is applied at every agreement tier, and unanimous decisions are reviewed with the same procedural rigour as ambiguous majority cases. Verification is conducted by two expert review panels comprising ten members in total, selected through purposive sampling on the basis of domain expertise, as shown in Figure 3.

**Figure 3.** Human-in-the-loop verification workflow.

The first panel consists of five HR professionals from the computer and technology sector, each with over ten years of experience in recruitment and talent management. Their primary responsibility is to validate the industry relevance of the skill-to-job assignments, ensuring the ontology reflects contemporary hiring practices and professional competency standards. During review, each member assessed whether a proposed mapping (i) correctly matched the functional meaning of the knowledge domain to the corresponding skill group,

(ii) was consistent with the professional skill vocabulary used in industry job postings, and (iii) did not inflate structural similarity for courses unrelated to computing work.

The second panel comprised five senior university lecturers with over 15 years of experience in computer science education and curriculum design. They assessed the pedagogical integrity of the knowledge-to-skill mappings in terms of semantic fit, domain relevance, and hierarchical consistency, ensuring that transcript-derived evidence was interpreted appropriately within an educational context. Their review criteria included whether a mapping (i) aligned with standard curriculum taxonomy in computer science and information technology programmes, (ii) correctly reflected the learning outcomes of the assigned knowledge domain, and (iii) preserved the intended separation between computing-oriented and arts-oriented competency paths.

The verification process is summarised in Table 9. Both panels reviewed the initial LLM assertions independently. Discrepancies and possible instances of semantic drift were recorded in a central revision log. Where inter-panel disagreement remained, a joint discussion was conducted to reach a documented final decision, supporting an ontology structure that is both academically grounded and professionally relevant.

Table 9. Human-in-the-loop verification protocol for validation of LLM-induced ontology structures in Phase 2.

Checkpoint	Objective	Procedure	Output Artefact
Semantic validation	Confirm that all mappings reflect the intended meaning of their definitions and that semantic drift is identified and corrected.	Both panels independently review knowledge-to-skill and skill-to-job assignments, flagging incorrect or ambiguous placements.	Approved mapping list and correction log with documented rationale for each revision.
Structural validation	Ensure that all ontology assertions are logically coherent and suitable for reasoning and path auditing in Phase 4.	Validated assertions are loaded into Protégé [32]; panels check for modelling inconsistencies and structural misplacements.	Validated ontology snapshot and a structured issue list with corrective actions where applicable.
Correction and traceability	Maintain transparent and reproducible documentation of all manual edits applied to LLM-generated outputs.	Each modification is recorded as an explicit edit entry comprising the original label, the revised label, and the expert's justification.	Traceable revision record supporting reproducibility auditing and iterative prompt refinement.

This design intentionally favours traceability and auditable hierarchy over full semantic coverage. Consequently, the resulting mappings should be interpreted as controlled analytical assignments rather than complete representations of all possible relationships among knowledge domains, skills, and job categories.

At the conclusion of Phase 2, the framework produces a HITL-validated ontology backbone linking O*NET-grounded knowledge nodes, LLM-induced skill groups with explicit definitions, and skill-to-job category assignments. This backbone supports semantic augmentation in Phase 3 and PCI-based multi-level path auditing in Phase 4.

The complete Phase 2 procedure, including ensemble-based induction, majority voting, HITL verification, and ontology materialisation, is summarised in Algorithm 2.

Algorithm 2: LLM-Ensemble Ontology Induction and HITL Validation (Phase 2)

Input: 22 O*NET knowledge domains $\mathcal{K}$; LLM ensemble $\mathcal{E} = \{\text{LLM}_1, \dots, \text{LLM}_5\}$
Output: HITL-validated ontology $\mathcal{O}$ with levels L_1, L_2, L_3
// Step 1: Induce L_2 skill groups
foreach $\text{LLM}_e \in \mathcal{E}$ **do**
 | Propose skill groups $\mathcal{SG}_e$ from $\mathcal{K}$ using constrained prompt;
 $\mathcal{SG} \leftarrow \text{MajorityVote}(\mathcal{SG}_1, \dots, \mathcal{SG}_5);$
// Step 2: Map $L_3 \rightarrow L_2$
foreach $k \in \mathcal{K}$ **do**
 | **foreach** $\text{LLM}_e \in \mathcal{E}$ **do**
 | Assign k to one group in $\mathcal{SG}$;
 $\text{assign}_{L_3 \rightarrow L_2}(k) \leftarrow \text{MajorityVote}(\dots);$
// Step 3: Map $L_2 \rightarrow L_1$
foreach $sg \in \mathcal{SG}$ **do**
 | **foreach** $\text{LLM}_e \in \mathcal{E}$ **do**
 | Assign sg to :Computer_Job or :Art_Job;
 $\text{assign}_{L_2 \rightarrow L_1}(sg) \leftarrow \text{MajorityVote}(\dots);$
// Step 4: HITL verification
foreach $\text{mapping } m \in \text{assign}_{L_3 \rightarrow L_2} \cup \text{assign}_{L_2 \rightarrow L_1}$ **do**
 | Expert reviews m ; corrects or confirms;
 | Record edit (original label, revised label, justification);
// Step 5: Materialise into ontology
 Encode accepted mappings as `rdfs:subClassOf` axioms in $\mathcal{O}$;
return $\mathcal{O}$;

4.5. Phase 3: Semantic Augmentation and Node Representation

Because LLM-generated labels may vary in wording across runs, Phase 3 decouples the human-readable label from the computational representation. Each ontology node is represented by a node description, while each textual evidence unit is represented by an augmented textual description. Together, these descriptions provide a controlled and contextually enriched basis for embedding-based similarity computation. The validated hierarchy from Phase 2 is then used alongside these representations for path auditing and competency ranking in later phases.

4.5.1. Semantic Augmentation Process

Semantic augmentation is applied to three types of textual units: ontology nodes, job-skill atoms, and student course records. For ontology nodes, O*NET knowledge domains are grounded in their standardised definitions, while LLM-induced skill-group nodes are represented using their HITL-validated descriptions. This separates stable external reference definitions from induced intermediate concepts.

For job-side units, each extracted skill atom is augmented in the context of the full job description to clarify its practical application and implied competency expectations. For course-side units, course titles and descriptions are expanded into competency-oriented statements that describe the knowledge and skills evidenced by each course. The augmentation prompts instruct the LLMs to consider labour-market, educational, and information-technology perspectives when expanding skills and course descriptions. This process is intended to shift similarity computation from surface-level keyword overlap toward functional meaning.

4.5.2. Embedding-Based Node Representation

After augmentation, each ontology node is represented by a node description (D_n), and each textual evidence unit is represented by an augmented textual description (T_u). These texts are encoded using SBERT [3]—specifically, the all-MiniLM-L6-v2 model—to produce fixed vector representations for subsequent similarity computation. Equations (1) and (2) define the embedding functions for ontology nodes and evidence units, respectively:

$$\vec{V}_n = f_{\text{SBERT}}(D_n), \quad n \in L_1 \cup L_2 \cup L_3, \quad (1)$$

$$\vec{V}_u = f_{\text{SBERT}}(T_u), \quad u \in \mathcal{U}. \quad (2)$$

In Equations (1) and (2), $f_{\text{SBERT}}(\cdot)$ denotes the all-MiniLM-L6-v2 Sentence-BERT encoder, D_n denotes the textual description used to represent ontology node n , and T_u denotes the augmented textual description of an evidence unit (u). Set $\mathcal{U}$ includes both job-skill atoms and student course records. For knowledge nodes (L_3), D_n is derived from the corresponding O*NET knowledge definition; for skill-group nodes (L_2), it is derived from the HITL-validated LLM description; and for job-category nodes (L_1), it is derived from the job-category definition used in Phase 2. Job-skill atoms are augmented using the skill item and its job-posting context, whereas student courses are represented using course descriptions expanded into competency-oriented statements.

At the conclusion of Phase 3, the framework produces two SBERT-based representation spaces: ontology-node embeddings ($\{\vec{V}_n\}$) derived from node descriptions and evidence-unit embeddings ($\{\vec{V}_u\}$) derived from augmented textual descriptions. These representations provide the computational basis for course-to-knowledge matching and PCI-based hierarchical scoring in Phase 4 by enabling consistent cosine-similarity comparisons across ontology levels and evidence types.

Algorithm 3 summarises the complete Phase 3 procedure, including semantic augmentation using the LLM ensemble and SBERT-based encoding of ontology nodes and evidence units.

Algorithm 3: Semantic Augmentation and SBERT Encoding of Ontology Nodes and Evidence Units (Phase 3)

Input: Ontology $\mathcal{O}$; job-skill atoms $\mathcal{A}_{\text{job}}$; course atoms $\mathcal{A}_{\text{stud}}$; LLM

Output: Node embeddings $\{\vec{V}_n\}$; evidence-unit embeddings $\{\vec{V}_u\}$

// Embed all ontology nodes

foreach node $n \in L_1 \cup L_2 \cup L_3$ **do**

$D_n \leftarrow$ definition/description from $\mathcal{O}$;

$\vec{V}_n \leftarrow f_{\text{SBERT}}(D_n)$;

// Augment and embed job-skill atoms

foreach atom $a = (\text{Job_ID}, \text{Job_Type}, s) \in \mathcal{A}_{\text{job}}$ **do**

$T_a \leftarrow \text{LLM}(s + \text{job context})$;

$\vec{V}_a \leftarrow f_{\text{SBERT}}(T_a)$;

// Augment and embed student course records

foreach record $r = (\text{Stu_ID}, \text{Course}, \text{Credits}, \text{Grade}) \in \mathcal{A}_{\text{stud}}$ **do**

$T_r \leftarrow \text{LLM}(\text{Course} \rightarrow \text{competency statement})$;

$\vec{V}_r \leftarrow f_{\text{SBERT}}(T_r)$;

return $\{\vec{V}_n\}, \{\vec{V}_u\} = \{\vec{V}_a\} \cup \{\vec{V}_r\}$;

4.6. Phase 4: Multi-Level Structural Scoring

Matching a course only to a knowledge node at L_3 is insufficient because high leaf-level similarity does not necessarily imply alignment with the corresponding skill group (L_2) or job category (L_1). Therefore, Phase 4 evaluates each course along the locked ontology path from L_3 to L_2 and L_1 . The resulting score is later used in Phase 5 to down-weight matches that appear relevant at the knowledge level but lack support from higher levels of the hierarchy.

4.6.1. Hierarchical Mapping Process

Hierarchical course positioning is performed in three steps: entry-point selection, branch locking, and multi-level re-scoring.

- Step 1: Entry-point selection at L_3 . Given a course embedding ($\vec{V}_i$), the entry node ($k^* \in L_3$) is identified as the knowledge node with the highest cosine similarity relative to the course. Equation (3) defines the top-1 selection rule and the corresponding leaf-level similarity score (S_3):

$$k^* = \arg \max_{k \in L_3} \cos(\vec{V}_i, \vec{V}_k), \quad S_3 = \cos(\vec{V}_i, \vec{V}_{k^*}). \quad (3)$$

Here, $\vec{V}_i$ is the embedding of course i as an evidence unit defined in Equation (2), and $\vec{V}_k$ is the embedding of knowledge node k defined in Equation (1). The selected node (k^*) serves as the starting point for all subsequent hierarchical checks.

- Step 2: Structural branch locking. Once k^* is determined, the corresponding ontology path is locked by retrieving its parent nodes directly from the ontology structure. Equation (4) defines the parent retrieval for the L_2 and L_1 nodes:

$$g^* = \text{parent}_{L_2}(k^*), \quad j^* = \text{parent}_{L_1}(g^*), \quad (4)$$

where g^* denotes the skill-group node at L_2 and j^* denotes the job-category node at L_1 . This locking step constrains consistency evaluation to a single well-defined hierarchical path rather than comparing the course against all possible cross-level combinations.

- Step 3: Multi-level re-scoring. The course embedding ($\vec{V}_i$) is re-evaluated against the locked parent nodes to obtain cosine-based contextual support scores at the parent levels. Equation (5) defines the parent-level similarity scores (S_2 and S_1):

$$S_2 = \cos(\vec{V}_i, \vec{V}_{g^*}), \quad S_1 = \cos(\vec{V}_i, \vec{V}_{j^*}). \quad (5)$$

Together, S_1 , S_2 , and S_3 capture alignment across the locked hierarchical path.

4.6.2. Path Consistency Index

The three similarity scores obtained from the hierarchical mapping process are consolidated into a structural confidence measure called the PCI. As defined in Equation (6), the PCI is the arithmetic mean of the level-wise similarity scores:

$$\text{PCI}_i = \frac{S_1 + S_2 + S_3}{3}. \quad (6)$$

A high PCI_i indicates that the course is not only similar to its matched knowledge node at L_3 but is also supported by the broader semantic context at the skill-group level (L_2) and job-category level (L_1). When S_3 is high but S_1 and S_2 are low, the course appears relevant in isolation but lacks hierarchical support, which may indicate a semantic mismatch caused by ambiguous course descriptions, broad elective content, or weak alignment with the

intended job path. By down-weighting such cases, the PCI provides a more structurally informed basis for competency assessment than leaf-level similarity alone.

The PCI_i value is propagated to Phase 5 as a continuous weighting signal in the job-conditioned scoring model so that courses with stronger hierarchical support contribute more to the final competency score. Rather than hard filtering, the framework retains courses with weak contextual support but assigns them a lower contribution than courses with consistent alignment across all three ontology levels.

At the conclusion of Phase 4, each course is assigned a locked ontology path, level-wise similarity scores (S_1, S_2, S_3), and a continuous PCI_i value. These outputs provide the structural weighting signals used in Phase 5 to compute the final competency score.

Algorithm 4 summarises the complete Phase 4 procedure, including ontology-path locking, level-wise cosine-similarity computation, and PCI calculation.

Algorithm 4: Ontology-Path Locking and Path Consistency Index Computation (Phase 4)

Input: Node embeddings $\{\vec{V}_n\}$; evidence-unit embeddings $\{\vec{V}_u\}$; ontology $\mathcal{O}$

Output: Locked paths $\mathcal{P}$; PCI scores $\{PCI_u\}$

foreach evidence unit $u \in \mathcal{A}_{job} \cup \mathcal{A}_{stud}$ **do**

 // Lock best L_3 node

$k^*(u) \leftarrow \arg \max_{k \in L_3} \cos(\vec{V}_u, \vec{V}_k);$

 Retrieve locked path: $k^*(u) \xrightarrow{\text{subClassOf}} sg^*(u) \xrightarrow{\text{subClassOf}} jc^*(u)$ from $\mathcal{O}$;

 // Level-wise similarities

$S_3 \leftarrow \cos(\vec{V}_u, \vec{V}_{k^*(u)});$

$S_2 \leftarrow \cos(\vec{V}_u, \vec{V}_{sg^*(u)});$

$S_1 \leftarrow \cos(\vec{V}_u, \vec{V}_{jc^*(u)});$

 // Path Consistency Index

$PCI_u \leftarrow (S_1 + S_2 + S_3) / 3;$

$\mathcal{P}(u) \leftarrow (k^*(u), sg^*(u), jc^*(u), S_1, S_2, S_3, PCI_u);$

return locked paths $\mathcal{P}$ and scores $\{PCI_u\}$;

4.7. Phase 5: Data-Driven Scaling and Final Evaluation

Phase 5 integrates academic performance, structural alignment from Phase 4, and job-side structural relevance signals into a job-conditioned competency score. Because grades and credits come directly from student transcripts and PCI_i is computed in Phase 4, this phase first defines a job-side importance factor, then computes the Total Accumulated Competency Score (TACS) and its normalised ranking variant.

4.7.1. Job Importance Factor

The job importance factor ($IF_{c,k}$) quantifies the job-side relevance of knowledge node k for job type c . It is derived from atomic job-skill records through a three-step procedure that combines requirement share, job-type specificity, and within-job normalisation. This design ensures that the factor reflects not only how frequently a node appears in a given job type but also how distinctively that node characterises that job type relative to others.

Step 1: Requirement mass. For each job type (c) and knowledge node (k), the requirement mass ($m_{c,k}$) accumulates the job-side path consistency scores of all atomic job-skill records (r) whose locked path passes through node k :

$$m_{c,k} = \sum_{r \in \mathcal{R}_{c,k}} PCI_r, \quad (7)$$

where $\mathcal{R}_{c,k}$ is the set of job-skill records of type c mapped to knowledge node k and PCI_r is the job-side path consistency score of record r . Nodes not observed in a given job type are assigned $m_{c,k} = 0$ rather than being excluded so that the full grid of job types and knowledge nodes is preserved.

Step 2: Requirement share and job-type specificity. The requirement share ($\text{REQ}_{c,k}$) normalises $m_{c,k}$ within job type c :

$$\text{REQ}_{c,k} = \frac{m_{c,k}}{\sum_{k' \in K_c} m_{c,k'}}, \quad (8)$$

where K_c is the set of L_3 nodes under the L_1 category associated with job type c .

The job-type-specific emphasis ($\text{Spec}_{c,k}$) measures how much node k is over-represented in job type c relative to its mean requirement share ($\overline{\text{REQ}}_k$) across all job types (C):

$$\overline{\text{REQ}}_k = \frac{1}{|C|} \sum_{c' \in C} \text{REQ}_{c',k}, \quad (9)$$

$$\text{Spec}_{c,k} = \log \left(1 + \frac{\max(0, \text{REQ}_{c,k} - \overline{\text{REQ}}_k)}{\varepsilon + \overline{\text{REQ}}_k} \right), \quad (10)$$

where $\varepsilon > 0$ is a small smoothing constant that prevents division by zero. A node with $\text{REQ}_{c,k} > \overline{\text{REQ}}_k$ receives a positive specificity score; a node that appears equally across all job types receives $\text{Spec}_{c,k} \approx 0$.

Step 3: Normalised importance factor. The importance factor is the product of requirement share and specificity, normalised within each job type so that the highest-weighted node receives a value of one:

$$\text{IF}_{c,k} = \frac{\text{REQ}_{c,k} \times \text{Spec}_{c,k}}{\max_{k'} (\text{REQ}_{c,k'} \times \text{Spec}_{c,k'})}. \quad (11)$$

Therefore, a high $\text{IF}_{c,k}$ indicates that knowledge node k is not only frequently observed in job type c (high $\text{REQ}_{c,k}$) but also specifically emphasised by that job type relative to others (high $\text{Spec}_{c,k}$). The factor is a dataset-conditioned structural relevance signal derived from observed job-posting evidence and should not be interpreted as a claim about general labour-market demand.

4.7.2. TACS Computation

For a target job type ($c \in \{\text{Data_Scientist}, \text{Data_Analyst}, \text{Data_Engineer}\}$), each transcript record (i) is mapped to a knowledge node ($k(i) \in L_3$) and assigned a structural score (PCI_i) from Phase 4. Equation (12) defines TACS_c as the weighted accumulation of job-conditioned competency evidence:

$$\text{TACS}_c = \sum_{i=1}^n \left(G_i \times Cr_i \times \text{PCI}_i \times \text{IF}_{c,k(i)} \right), \quad (12)$$

where G_i is the numeric grade of transcript record i , Cr_i is its credit weight, PCI_i is the student-side structural alignment score from Phase 4, $\text{IF}_{c,k(i)}$ is the job-side structural relevance factor defined in Section 4.7.1, and n is the number of transcript records included in the computation. This formulation captures accumulated job-conditioned competency evidence prior to normalising for differences in transcript length and credit volume.

4.7.3. Normalised TACS and Candidate Ranking

Because students differ in transcript length and credit volume, the raw $TACS_c$ is normalised by the total relevance weight associated with the transcript records. Equation (13) defines the normalised score ($NTACS_c$) and its record-level relevance weight (w_i):

$$NTACS_c = \frac{\sum_{i=1}^n (G_i \times w_i)}{\sum_{i=1}^n w_i}, \quad w_i = Cr_i \times PCI_i \times IF_{c,k(i)}. \quad (13)$$

where w_i is the job-conditioned relevance weight of transcript record i that combines learning intensity (Cr_i), student-side structural validity (PCI_i), and job-side structural relevance ($IF_{c,k(i)}$). Unlike $TACS_c$, which represents raw accumulated evidence, $NTACS_c$ supports fairer comparison across students by normalising against the total relevance weight. Candidate ranking is based on $NTACS_c$, while $TACS_c$ is retained as the un-normalised accumulated score.

To complement this grade-sensitive ranking score, the framework also computes a relevant volume measure that captures the breadth of structurally supported coursework independently of grades. Equation (14) defines this volume:

$$RelVol = \sum_{i=1}^n (Cr_i \times PCI_i). \quad (14)$$

where $RelVol$ measures the total credit volume supported by the locked ontology path, regardless of grade performance or job-side relevance. It is used only as a supplementary structural volume indicator.

4.7.4. Semantic Traceability and Explainability

To make ranking outcomes interpretable, the framework records a course-level evidence-relevance signal that combines student-side structural alignment with job-side structural relevance. Equation (15) defines this signal (E_i):

$$E_i = PCI_i \times IF_{c,k(i)}. \quad (15)$$

where E_i represents the relevance of transcript record i to the target job type before grade and credit weighting. Aggregating E_i , optionally weighted by course credits, by knowledge node or skill group produces a semantic traceability map that explains which parts of a student's transcript provide the strongest semantic evidence for a target job type.

At the conclusion of Phase 5, the framework produces job-conditioned rankings, normalised competency scores, relevant volume measures, and course-level traceability signals for each target job type.

Algorithm 5 summarises the complete Phase 5 procedure, including job-type importance factor computation, TACS and NTACS calculation, candidate ranking, and semantic traceability signal generation.

A complete worked example tracing Student A through all five phases of the O4CM framework is provided in Appendix A.

Algorithm 5: Job-Conditioned Competency Scoring and Candidate Ranking (Phase 5)

Input: Locked paths $\mathcal{P}_{\text{job}}, \mathcal{P}_{\text{stud}}$; PCI scores; grades G_i ; credits Cr_i ; small constant ε

Output: $IF_{c,k}$; $NTACS_c(s)$ for all students s ; ranked lists $\mathcal{R}_c$; traceability signals $\{E_i\}$

// Step 1: Compute job-side Importance Factor

foreach job type c and node k **do**

$m_{c,k} \leftarrow \sum_{r \in \mathcal{R}_{c,k}} \text{PCI}_r$;

$\text{REQ}_{c,k} \leftarrow m_{c,k} / \sum_{k'} m_{c,k'}$;

foreach node k **do**

$\overline{\text{REQ}}_k \leftarrow \frac{1}{|\mathcal{C}|} \sum_{c'} \text{REQ}_{c',k}$;

foreach job type c **do**

$\text{Spec}_{c,k} \leftarrow \log \left(1 + \frac{\max(0, \text{REQ}_{c,k} - \overline{\text{REQ}}_k)}{\varepsilon + \overline{\text{REQ}}_k} \right)$;

Normalise: $IF_{c,k} \leftarrow (\text{REQ}_{c,k} \cdot \text{Spec}_{c,k}) / \max_{k'} (\text{REQ}_{c,k'} \cdot \text{Spec}_{c,k'})$;

// Step 2: Score and rank all students

foreach student s and job type c **do**

$\text{TACS}_c(s) \leftarrow \sum_i G_i \cdot Cr_i \cdot \text{PCI}_i \cdot IF_{c,k(i)}$;

$w_i \leftarrow Cr_i \cdot \text{PCI}_i \cdot IF_{c,k(i)}$;

$\text{NTACS}_c(s) \leftarrow \sum_i (G_i \cdot w_i) / \sum_i w_i$;

$E_i \leftarrow \text{PCI}_i \cdot IF_{c,k(i)}$;

// traceability signal

foreach job type c **do**

$\mathcal{R}_c \leftarrow$ students sorted by NTACS_c (descending);

return $IF_{c,k}, \text{NTACS}_c, \mathcal{R}_c, \{E_i\}$;

5. Results

The five-phase O4CM framework described in Section 4 is evaluated along three dimensions: the reliability of the induced ontology backbone, the discriminative value of semantic representation and PCI-based structural scoring, and the ranking behaviour and interpretability of the final student-to-job matching output.

The results are organised around the three contributions stated in the Introduction. Section 5.3 supports the first contribution by examining whether the LLM-assisted and HITL-verified process produces a coherent ontology backbone for multi-level competency mapping. Sections 5.4 and 5.5 support the second contribution by evaluating the design of the semantic representation and the discriminative value of PCI-based structural scoring. Sections 5.6 and 5.7 support the second and third contributions by analysing component-level effects, job-conditioned ranking behaviour, and the traceability of ranking outputs to course-level evidence.

5.1. Dataset and Experimental Setup

The evaluation draws on the datasets described in Section 3. The job-posting corpora provide job-side evidence, while anonymised student transcript records provide academic learning evidence. All experiments use the same HITL-validated ontology backbone from Phase 2, the same semantic augmentation and SBERT representation process from Phase 3, and the same PCI-based structural scoring procedure from Phase 4.

The full transcript dataset contains 430 students and is used to compute course-level ontology mappings and PCI_i values. For group-level structural validation, a subset of 244 students is used, comprising 185 computing students and 59 Visual Arts students. This subset provides the clearest contrast between computing-oriented and arts-oriented

curricula for evaluation of ontology-path separation and is not used for parameter tuning. Ablation and ranking analyses are conducted on a stratified 94-student subsample, comprising 73 computing students and 21 Visual Arts students, to support controlled comparison across methods and framework variants. All subsets are drawn from the same population described in Table 7.

The evaluation reports four complementary types of evidence. The first concerns ontology induction and expert validation. The second examines whether semantic representations provide a richer basis for matching than surface labels. The third evaluates whether PCI-based scoring separates contrasting academic groups under different ontology paths. The fourth uses comparison methods, ablation variants, ranking measures, and traceability evidence to analyse the behaviour of the final framework.

5.2. Comparison Methods and Ablation Variants

This subsection defines two types of experimental references. The comparison methods represent progressively stronger alternatives for student-to-job ranking, from grade-only ranking to the full proposed framework. The ablation variants remove individual components from the framework to test whether each component contributes to the final ranking behaviour.

Comparison Methods

Five comparison methods are evaluated. These methods are not presented as full competing systems from prior work. Rather, they serve as controlled reference points that reflect increasing levels of ranking and matching complexity, moving from grade-only ranking and lexical matching to embedding-based semantic similarity and the full proposed framework. Table 10 summarises the five methods and highlights which components are enabled in each variant, including semantic encoding, usage of an ontology structure, multi-level PCI, job-side $IF_{c,k}$ weighting, and the final ranking basis.

Table 10. Comparison methods used in the ranking evaluation. ✓ indicates that the component is applied, ◦ indicates partial application, and × indicates that the component is absent.

Method	Semantic Encoding	Ontology Structure	Multi-Level PCI	$IF_{c,k}$ Weighting	Ranking Basis
M1: GPA Ranking	×	×	×	×	Cumulative GPA
M2: Keyword Matching	×	×	×	×	String overlap
M3: Flat SBERT	✓	×	×	×	Flat cosine similarity
M4: L_3 -Only + IF	✓	◦	×	✓	$S_{3,i} \times IF_{c,k(i)}$
M5: Proposed Framework	✓	✓	✓	✓	NTACS _c

M1: GPA Ranking.

Candidates are ranked by cumulative GPA without considering the semantic content or job relevance of individual courses. This method represents a conventional grade-based reference strategy where academic performance is used as an aggregate indicator of student achievement. However, GPA alone may obscure differences in learning patterns and competency relevance that are important for job-specific evaluation [33].

M2: Keyword Matching.

Competency alignment is estimated using normalised string overlap between course names and job-skill terms. This method represents a surface-level lexical matching strategy for job-requirement analysis. Although keyword-based search can identify explicit skill terms in job texts, it relies on predefined keyword lists and must be manually updated when new requirements appear [17]. Thus, M2 remains limited when courses and job postings express the same competency using different wording.

M3: Flat SBERT.

Course and job-skill representations are encoded using the same SBERT model as the proposed framework [3]. This method uses cosine similarity between sentence embeddings as a flat semantic matching strategy, without ontology-based branch locking or multi-level PCI scoring. Prior resume-screening research shows that SBERT can rank candidate profiles against job descriptions more effectively than keyword-based matching by capturing contextual semantic similarity [34]. However, this type of matching remains structurally flat: it compares textual representations directly but does not verify whether the matched competency is coherent across knowledge, skill-group, and job-category levels. Therefore, M3 isolates the contribution of semantic encoding before adding the ontology-based verification mechanism proposed in O4CM.

M4: L_3 -Only + IF.

This method uses the top-matched knowledge node at L_3 and applies the job-side structural relevance factor ($IF_{c,k}$) defined in Equation (11), but it does not use the full three-level PCI_i score. Each course is scored using the leaf-level similarity score (S_3) defined in Equation (3) rather than the full locked-path score. This framework-derived comparator tests whether leaf-level matching is sufficient when the job-side relevance signal is retained.

M5: Proposed Framework.

The proposed framework combines augmented semantic representations, ontology-based branch locking, PCI-based multi-level structural scoring, and job-side $IF_{c,k}$ weighting. Candidate ranking is based on $NTACS_c$ as defined in Equation (13).

5.3. Ontology Induction and Expert Validation

This subsection reports the Phase 2 ontology induction results. The analysis focuses on the degree of cross-model agreement during LLM-assisted induction and the role of HITL validation in converting candidate mappings into an expert-confirmed ontology backbone.

The induction process covers three tasks: skill-group induction, Knowledge Domain (L_3)-to-Skill Group (L_2) mapping, and Skill Group (L_2)-to-Job Category (L_1) assignment. Figure 4 presents the validated ontology backbone used in the subsequent semantic augmentation and PCI-based structural scoring phases. The validated L_2 layer consists of seven skill groups: Cognitive Skills, Technical Skills, Creative Skills, Communication Skills, Social and Interpersonal Skills, Psychomotor Skills, and Affective Skills.

To summarize the materialized ontology backbone, Table 11 reports the number of nodes and subclass links retained after HITL verification. The table reports the analytical backbone used for scoring, not all auxiliary OWL entities contained in the ontology file.

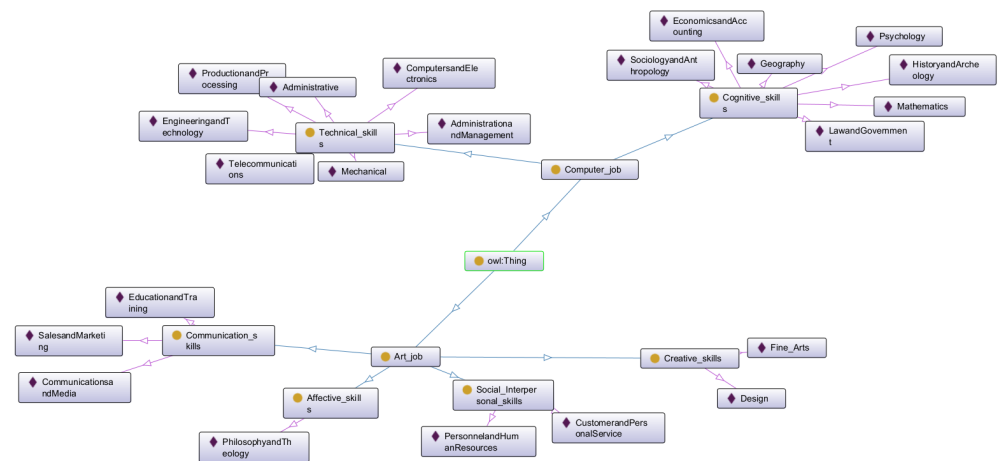


Figure 4. Expert-validated ontology structure produced by Phase 2, visualized in Protégé [32]. The two root nodes form the L_1 layer, seven skill-group nodes form the L_2 layer, and 22 O*NET knowledge domains form the L_3 layer. Arrows denote `subclassOf` relations from child to parent nodes.

Table 11. Summary of the expert-confirmed ontology backbone produced in Phase 2.

Ontology Component	Count	Source	Validation Status
Job categories (L_1)	2	Defined by the study scope.	HITL-confirmed.
Skill groups (L_2)	7	LLM-induced from O*NET knowledge definitions.	HITL-confirmed.
Knowledge domains (L_3)	22	O*NET knowledge taxonomy.	Used as fixed semantic anchors.
Knowledge-to-skill subclass links	22	LLM-proposed and expert-reviewed.	Materialised after HITL verification.
Skill-to-job subclass links	7	LLM-proposed and expert-reviewed.	Materialised after HITL verification.

5.3.1. LLM Ensemble Agreement

The five-model LLM ensemble produced candidate knowledge-to-skill mappings for the 22 O*NET knowledge domains. Agreement was measured as the fraction of models that assigned the same skill-group label to each knowledge node. Table 12 summarises the observed agreement distribution.

Table 12. Agreement distribution of the five-model LLM ensemble for Knowledge Domain $\rightarrow$ Skill Group induction ($N = 22$).

Agreement Level	Consensus	Count	Percentage (%)	Interpretation
Unanimous	5/5	17	77.27	High observed agreement
High Majority	4/5	4	18.18	Strong but not unanimous agreement
Simple Majority	3/5	1	4.55	Ambiguous case requiring review

The ensemble produced high observed agreement, with 21 of 22 knowledge nodes reaching at least 4/5 consensus. This corresponds to $21/22 = 95.45\%$ of knowledge domains achieving high-consensus agreement, which is the figure reported in the Abstract. This result indicates that the prompt constraints produced consistent outputs for most mappings. However, agreement is treated only as a preliminary signal. As shown by the expert review cases below, even unanimous agreement may still require correction when the assignment is not functionally aligned with the target domain.

5.3.2. HITL Validation Outcomes

Table 13 summarises how the LLM-generated candidate mappings were handled during HITL validation. The purpose of expert validation was not to report model accuracy against an external gold standard but to determine whether each candidate mapping was acceptable for ontology materialisation.

Table 13. HITL validation summary for LLM-induced knowledge-to-skill mappings.

Agreement Level	Consensus	Count	Expert Review Action	Final Validation Status
Unanimous	5/5	17	Reviewed for domain fit despite full model agreement.	Mostly confirmed; one unanimous case was corrected after expert review.
High Majority	4/5	4	Reviewed with attention to the dissenting label.	HITL-confirmed after review.
Simple Majority	3/5	1	Escalated for detailed expert judgement.	HITL-corrected or confirmed with documented rationale.

The validation results show that LLM agreement is useful for prioritising review effort but is not sufficient for final ontology acceptance. Therefore, all materialised mappings were confirmed through expert review before being used in Phase 3 and Phase 4.

5.3.3. Representative Expert Correction Cases

Three representative cases illustrate why HITL review is necessary, even when LLM outputs appear plausible.

Case 1: Sociology and Anthropology.

This node produced the lowest agreement among the 22 knowledge domains. Most models associated it with *Social_Interpersonal_Skills* due to the surface association with the term “social”, while other models favoured *Cognitive_Skills*. The expert panels assigned it to *Cognitive_Skills* because, in computer-job contexts such as data science and UX research, the domain functions primarily as an analytical framework for behavioural analysis and user modelling.

Case 2: Administration and Management.

All five models assigned this domain to *Cognitive_Skills*, interpreting management as abstract decision-making. The expert panels reassigned it to *Technical_Skills* because, in technology-sector practice, this domain is often expressed through procedural and tool-mediated competencies such as project management methods, Agile practice, Scrum, and IT operations frameworks. This case demonstrates that unanimous LLM agreement does not guarantee domain-correct placement.

Case 3: Philosophy and Theology.

This domain was judged to occupy a boundary between reasoning, ethics, and value-oriented judgement. The expert panels placed it under *Affective_Skills* and assigned it to the Art Job category. This placement prevents ethics-oriented or value-oriented transcript evidence from inflating similarity to computer-job paths during structural scoring.

At the conclusion of this validation stage, the framework produces an expert-confirmed ontology backbone that supports semantic augmentation in Phase 3 and PCI-based structural scoring in Phase 4.

5.4. Semantic Representation Diagnostics

This subsection provides a diagnostic comparison of the text sources used for embedding-based matching. The purpose is not to report a separate classification benchmark but to clarify why the proposed framework uses augmented evidence descriptions and ontology-node definitions rather than surface labels alone.

As shown in Table 14, surface-label matching is vulnerable to lexical sparsity because course titles and skill names often omit relevant competencies. Representing evidence units

with augmented descriptions and ontology nodes with definitions provides richer semantic context on both sides of the comparison. Table 15 illustrates this effect using representative examples from the evidence and ontology layers.

Table 14. Diagnostic comparison of text-source configurations for semantic representation.

Configuration	Text Sources	Diagnostic Interpretation
Name × Name	Evidence label matched against ontology label.	Compact but highly sensitive to surface-vocabulary mismatch.
Name × Definition	Evidence label matched against ontology definition.	Improves ontology-side context, but short evidence labels may still under-represent meaning.
Description × Name	Augmented evidence description matched against ontology label.	Adds evidence-side context but remains constrained by short ontology labels.
Description × Definition	Augmented evidence description matched against ontology definition.	Provides semantic context on both sides and is adopted in the proposed framework.

Table 15. Illustrative examples of added semantic context from augmented descriptions and node definitions.

Unit Type	Surface Label	Added Semantic Context
Academic course	Data Structures	Study and practical training related to data representation; data structures and design, including arrays, stacks, queues, linked lists, trees, and graphs; data sorting; data searching; and algorithm analysis.
Job requirement	SAS	A software suite that integrates advanced analytics, business intelligence, data management, and predictive modelling to empower data-driven decision-making.
Knowledge node	Computers and Electronics	Knowledge of circuit boards, processors, chips, electronic equipment, and computer hardware and software, including applications and programming.

These examples are used as diagnostic illustrations rather than independent performance evidence. Together, the configuration comparison and illustrative examples support the use of description-to-definition matching in the proposed framework.

5.5. PCI-Based Group Separation

This subsection evaluates whether the PCI-based structural score separates computing-oriented and arts-oriented curricula under the two ontology root paths. The analysis uses student-level credit-weighted PCI scores and compares the computing group with the Visual Arts control group. This evaluation assesses group-level structural separation rather than course-level classification accuracy.

For each student (s), the credit-weighted PCI score aggregates course-level PCI_i values using course credits as weights. Equation (16) defines the student-level score used in the group comparison:

$$PCI_s^{(w)} = \frac{\sum_{i \in \mathcal{C}_s} Cr_i \cdot PCI_i}{\sum_{i \in \mathcal{C}_s} Cr_i}, \quad (16)$$

where $\mathcal{C}_s$ is the set of courses taken by student s . The Mann–Whitney U test is used to compare student-level PCI distributions between the computing and Visual Arts groups because it is a rank-based test for evaluating whether one independent sample tends to produce larger values than another [35].

The positive group consists of 185 students from the four computing programmes, while the control group consists of 59 students from the B.F.A. Visual Arts programme.

Table 16 reports the group means, separation ratio, effect size, and significance level for each ontology path. To facilitate interpretation, we report the Separation Ratio (SR), defined as the ratio of the mean score of the computing group to the mean score of the Visual Arts group. Under the Computer Job path, $SR > 1$ indicates the expected direction, whereas under the Art Job path, $SR < 1$ indicates the expected direction.

Cohen's d is reported as a descriptive effect-size measure to support interpretation of group-separation magnitude.

Table 16. Mann–Whitney U test comparing the student-level credit-weighted PCI score defined in Equation (16) between computing students ($n = 185$) and Visual Arts students ($n = 59$) under each ontology path.

Ontology Path	Mean Score, Computing	Mean Score, Visual Arts	SR	Cohen's d	p -Value
Computer Job	0.2483	0.2185	1.136	+1.065	<0.001
Art Job	0.2147	0.3358	0.639	−2.005	<0.001

Both comparisons are significant, with large effects. Computing students achieve higher scores on the Computer Job path, while Visual Arts students achieve higher scores on the Art Job path. This bidirectional pattern supports the structural validity of the ontology backbone and indicates that PCI-based scoring captures meaningful curriculum-level differences between the two groups.

5.6. Ablation Study

The ablation study examines whether the major components of O4CM contribute to discriminative behaviour. Three components are tested: semantic augmentation, PCI-based multi-level structural scoring, and job-side $IF_{c,k}$ weighting. Four variants are evaluated by removing one component at a time while keeping the remaining components fixed:

- V1 Full Framework: Augmented representations, three-level PCI_i , and job-side $IF_{c,k}$ are all retained.
- V2 Without Augmentation: Raw course names are matched against raw node labels, while PCI_i and $IF_{c,k}$ are retained.
- V3 Without PCI: Augmented representations are retained, but the three-level PCI_i score is replaced by the leaf-level similarity score (S_3).
- V4 Without IF: Augmented representations and PCI_i are retained, but $IF_{c,k}$ is set to 1 for all knowledge nodes.

Two complementary metrics are used to quantify the impact of each ablation on group separation. The first is the IF-weighted relevant credit volume, which captures grade-free structural alignment weighted by job-side relevance. Equation (17) defines RCV_c as a credit-normalised accumulation of path-consistent, job-relevant coursework:

$$RCV_c = \frac{\sum_{i=1}^n (Cr_i \times PCI_i \times IF_{c,k(i)})}{\sum_{i=1}^n Cr_i}, \quad (17)$$

where Cr_i is the credit value of course record i , PCI_i is the course-level path consistency score from Phase 4, $IF_{c,k(i)}$ is the job-side importance factor for the mapped knowledge node ($k(i)$), and n is the number of transcript records.

The second metric is the grade-weighted competency score, which extends the previous measure by incorporating academic performance. Equation (18) defines GCS_c by additionally weighting each record by its numeric grade (G_i):

$$GCS_c = \frac{\sum_{i=1}^n (G_i \times Cr_i \times PCI_i \times IF_{c,k(i)})}{\sum_{i=1}^n Cr_i}, \quad (18)$$

where G_i denotes the numeric grade for course record i . Together, RCV_c and GCS_c separate the contributions of structure and market relevance (RCV) from the additional influence of academic performance (GCS), enabling a more interpretable ablation analysis.

For each variant and metric, SR is computed as in Section 5.5, using the ratio of mean scores between the computing group and the Visual Arts group. The expected direction is $SR > 1$ for the Computer Job path and $SR < 1$ for the Art Job path.

Table 17 shows that no single component maximises all outcomes. V2 produces higher Computer Job separation than the full framework, but it weakens the Art Job path, suggesting that raw labels may increase single-domain separation while reducing negative-control cross-domain discrimination. Therefore, semantic augmentation is not used solely to maximise separation on one path; rather, it helps reduce over-specialised surface matching and supports more stable semantic ranking behaviour across the two contrasting ontology paths evaluated in this study.

Table 17. Ablation study results. SR is the ratio of the mean computing student score to the mean Visual Arts student score. For the Computer Job path, $SR > 1$ indicates the expected direction; for the Art Job path, $SR < 1$ indicates the expected direction.

Variant	Configuration	Computer Job			Art Job	
		RCV SR	GCS SR	Cohen's d	RCV SR	GCS SR
V1 Full	Aug + PCI + IF	1.8633	1.6864	3.168	0.6267	0.5561
V2 Without Augmentation	Raw + PCI + IF	2.1586	1.8997	4.113	0.6990	0.6179
V3 Without PCI	Aug + S_3 + IF	1.7993	1.6389	3.589	0.8891	0.7864
V4 Without IF	Aug + PCI + IF = 1	1.2289	1.0962	5.461	0.5400	0.4824

The remaining ablations show complementary component effects. Removing $IF_{c,k}$ in V4 substantially weakens Computer Job separation, indicating the importance of job-side relevance weighting for data-domain ranking. Removing PCI in V3 mainly affects the Art Job path, where the SR moves closer to the neutral value, indicating that multi-level structural scoring is important in limiting cross-domain contamination. Overall, the full framework provides the most balanced behaviour across both ontology paths, even though it does not achieve the largest value on every individual metric.

5.7. Ranking Performance and Semantic Traceability

This subsection evaluates the final ranking behaviour of the proposed framework against the comparison methods defined in Section Comparison Methods. The analysis considers group-level separation, retrieval performance, sub-type sensitivity, and semantic traceability.

5.7.1. Group Separation Across Methods

SR quantifies how strongly each method separates the target group from the contrast group. For data-domain job types, SR is computed as the mean score of computing students

divided by the mean score of Visual Arts students. For the Visual Art job type, the ratio is reversed so that an SR above one still indicates the expected direction. Cohen's d is reported as a descriptive effect-size measure comparing the target and contrast groups under each method and job type. Table 18 reports the SR and Cohen's d for all five methods across the four job types.

Table 18. SR and Cohen's d for each method across four job types.

Job Type	M1: GPA Ranking	M2: Keyword Matching	M3: Flat SBERT	M4: L_3 -Only + IF	M5: Proposed Framework
Separation Ratio (SR)					
Data Scientist	0.892	2.590	1.238	1.821	1.673
Data Analyst	0.892	2.590	1.238	1.876	1.723
Data Engineer	0.892	2.590	1.238	1.887	1.798
Visual Art	1.121	1.881	1.268	1.169	1.728
Cohen's d					
Data Scientist	−0.990	+3.558	+5.566	+3.458	+3.187
Data Analyst	−0.990	+3.558	+5.566	+2.320	+2.433
Data Engineer	−0.990	+3.558	+5.566	+4.006	+3.984
Visual Art	+0.990	+3.326	+7.357	+1.394	+6.982

Table 18 shows that GPA ranking fails data-domain directionality because Visual Arts students have the highest mean GPA in the sample. Keyword matching and flat SBERT achieve correct broad-domain separation in several cases, but they do not distinguish among Data Scientist, Data Analyst, and Data Engineer requirements. The proposed framework is not always the highest-scoring method on every individual metric, but it is the only method that jointly satisfies the four operational criteria used in this evaluation: domain discrimination, sub-type sensitivity, negative-control cross-domain discrimination, and ontology-based interpretability.

5.7.2. Retrieval Performance and Ranking Consistency

Recall@ K and Hit Rate@ K evaluate whether relevant candidates appear near the top of the ranked list. Recall@ K measures the proportion of target-group students included in the top K , while Hit Rate@ K measures the percentage of top- K positions occupied by the target group. Table 19 reports results for $K \in \{10, 20, 50\}$.

Table 19. Recall@ K and Hit Rate@ K for computing students on data-domain job types and for Visual Arts students on the Visual Art job type. Hit Rate@ K reports the percentage of top- K positions occupied by target-group students.

Job Type	K	M1: GPA Ranking	M2: Keyword Matching	M3: Flat SBERT	M4: L_3 -Only + IF	M5: Proposed Framework
Recall@ K —computing students (data-domain roles, all three job types identical)						
Data-domain (DS/DA/DE)	10	0.068	0.137	0.137	0.137	0.137
	20	0.164	0.274	0.274	0.274	0.274
	50	0.466	0.685	0.685	0.685	0.685
Recall@ K —Visual Arts students (Visual Art job type)						
Visual Art	10	0.238	0.476	0.476	0.000	0.476
	20	0.381	0.952	0.952	0.333	0.952
	50	0.762	1.000	1.000	1.000	1.000

Table 19. Cont.

Job Type	K	M1: GPA Ranking	M2: Keyword Matching	M3: Flat SBERT	M4: L_3 -Only + IF	M5: Proposed Framework
Hit Rate@K—computing students (data-domain roles)						
Data-domain (DS/DA/DE)	10	50%	100%	100%	100%	100%
	20	60%	100%	100%	100%	100%
	50	68%	100%	100%	100%	100%
Hit Rate@K — Visual Arts students (Visual Art job type)						
Visual Art	10	50%	100%	100%	0%	100%
	20	40%	100%	100%	35%	100%
	50	32%	42%	42%	42%	42%

Several semantic methods reach the maximum possible Recall@K for data-domain roles because all top-K positions are occupied by computing students. In this setting, Recall@K is bounded by the number of computing students in the 94-student ranking subsample rather than by the top-K list alone. Thus, the main contrast is not data-domain retrieval but the Visual Art path, where L_3 -only matching performs poorly while the proposed framework remains robust. This supports the role of multi-level structural scoring in reducing cross-domain contamination.

5.7.3. Sub-Type Sensitivity and Overall Criteria

Sub-type sensitivity is evaluated in terms of whether a method produces distinct ranking behaviour across Data Scientist, Data Analyst, and Data Engineer roles. Methods without job-side $IF_{c,k}$ weighting tend to produce identical or near-identical patterns across the three data-domain roles, whereas methods using $IF_{c,k}$ preserve role-specific differences derived from the job-side structural relevance analysis.

Spearman's ρ is used as a supplementary summary measure of pairwise ranking agreement between methods. Across the data-domain job types, the rankings produced by M4 (L_3 -Only + IF) and M5 (proposed framework) show consistently high agreement ($\rho \geq 0.994$, $p < 10^{-89}$). This indicates that M5 preserves much of the semantic ranking structure produced by M4 while adding PCI-based structural safeguards. In contrast M1 (GPA ranking) shows weak or negative correlation with the semantic methods, suggesting that grade-only ranking produces substantially different candidate orderings.

Table 20 summarises method performance across four criteria. Domain discrimination is met when the expected SR direction is observed across job types. Sub-type sensitivity requires distinct scores across the three data-domain roles. Cross-domain robustness is assessed on the Visual Art path, and ontology-based interpretability is assessed in terms of whether scores can be traced to ontology nodes and paths.

The proposed framework is the only method satisfying all four criteria in this evaluation. This does not mean that it maximises every individual metric. Rather, its advantage is that it combines acceptable domain discrimination, role-specific sensitivity, robustness on the cross-domain Visual Art path, and traceable ontology-based interpretation within a single scoring framework.

Table 20. Summary of method performance across four operational criteria under the dataset and target-role setting evaluated in this study. ✓ indicates that the criterion is met; × indicates failure; △ indicates partial performance. The proposed framework is the only method satisfying all four criteria in this evaluation.

Criterion	M1: GPA Ranking	M2: Keyword Matching	M3: Flat SBERT	M4: L ₃ -Only + IF	M5: Proposed Framework
Domain discrimination	×	✓	✓	✓	✓
Sub-type sensitivity	×	×	×	✓	✓
Cross-domain robustness	△	✓	△	×	✓
Ontology-based interpretability	×	△	△	△	✓
Criteria met	0/4	2/4	2/4	2/4	4/4

5.7.4. Semantic Traceability Analysis

Using the course-level evidence-relevance signal (E_i) defined in Equation (15), the framework traces each ranking outcome back to course-level semantic evidence. Because E_i is computed before grade and credit weighting, it highlights the structural and job-side relevance of each course independently of the student's achieved grade.

For each student (s) and target job type (c), course-level evidence relevance is aggregated by skill group to form a competency profile. Equation (19) defines the skill-group profile value ($P_{s,c,g}$) as the credit-weighted mean evidence relevance within skill group g :

$$P_{s,c,g} = \frac{\sum_{i: g^*(i)=g} Cr_i \times E_i}{\sum_{i: g^*(i)=g} Cr_i}, \quad (19)$$

where $g^*(i) = \text{parent}_{L_2}(k(i))$ denotes the skill group assigned to course i through the branch-locking step and E_i is the course-level evidence-relevance signal defined in Equation (15). Each dimension of $\vec{P}_{s,c}$ represents the credit-weighted mean evidence relevance within a skill group. Therefore, the profile explains which ontology areas provide the strongest semantic evidence for the target job type before grade weighting.

Figure 5 presents Semantic Traceability Maps for four representative students from B.Sc. Computer Science, B.B.A. Business Information Systems, B.Eng. Computer Engineering, and B.F.A. Visual Arts, evaluated against the Data Scientist job type.

The computing students' profiles concentrate more strongly in Technical Skills and Cognitive Skills, which is consistent with their computing-oriented curricula. The Visual Arts profile places greater weight on art-oriented skill groups such as Creative Skills and Affective Skills. This contrast illustrates how the framework connects ranking behaviour to interpretable ontology-level evidence rather than only producing a numerical score.

Table 21 reports the five highest-ranked courses by credit-weighted evidence relevance ($Cr_i \times E_i$) for two contrasting students evaluated against the Data Scientist job type. The table should be interpreted as an explanation of semantic evidence relevance, not as a full decomposition of NTACS_c , because E_i is defined before grade weighting.

Student A's top courses map to knowledge nodes that are structurally relevant to the Data Scientist path in the job-side analysis. Student B's top courses map mainly to Fine Arts and Communications and Media, yielding lower evidence-relevance values for the Data Scientist path. This comparison illustrates how the framework provides traceable course-level evidence alongside the ranking output.

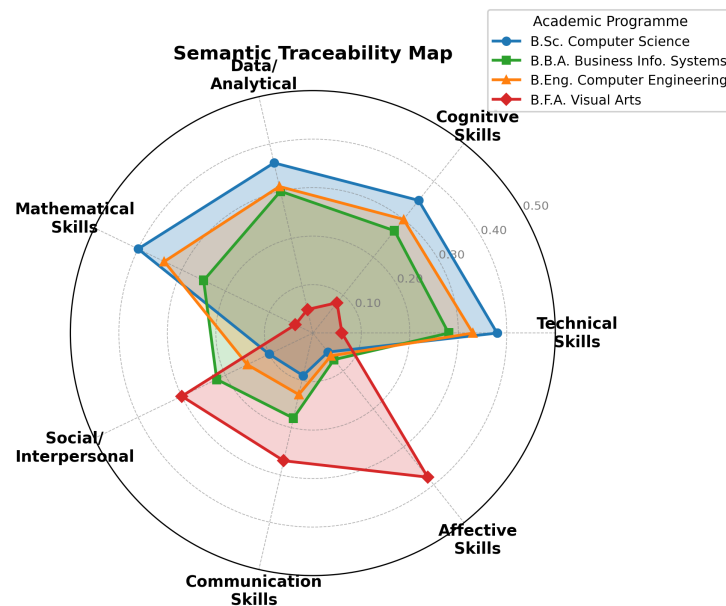


Figure 5. Semantic traceability maps for four representative students evaluated against the Data Scientist job type. Each axis corresponds to one of the seven skill groups at L_2 , and the shaded area reflects the credit-weighted mean evidence relevance ($P_{s,c,g}$) defined in Equation (19).

Table 21. Top-5 contributing courses by credit-weighted evidence relevance ($Cr_i \times E_i$) for two representative students evaluated against the Data Scientist job type. Cr_i denotes course credit, $k(i)$ denotes the mapped knowledge node at L_3 , PCI_i denotes the Path Consistency Index, and E_i is the pre-grade evidence-relevance signal defined in Equation (15).

Rank	Course Name	Mapped Knowledge Node ($k(i)$)	Cr_i	PCI_i	E_i
Student A: B.Sc. Computer Science (NTACSDS Rank 1 of 94)					
1	Machine Learning	Mathematics	3	0.481	0.342
2	Database Systems	Computers and Electronics	3	0.463	0.319
3	Data Mining	Mathematics	3	0.457	0.325
4	Algorithm Design	Mathematics	3	0.445	0.316
5	Software Engineering	Computers and Electronics	3	0.432	0.298
Student B: B.F.A. Visual Arts (NTACSDS Rank 76 of 94; top-ranked arts student)					
1	Art History and Theory	Fine Arts	3	0.361	0.094
2	Design Studio I	Fine Arts	3	0.344	0.089
3	Creative Media	Fine Arts	3	0.331	0.086
4	Visual Communication	Communications and Media	3	0.298	0.071
5	Photography	Fine Arts	3	0.287	0.074

Taken together, the results provide empirical support for the three contributions stated in the Introduction. First, the LLM ensemble and HITL validation results show that the semi-automated process can produce an expert-confirmed ontology backbone suitable for downstream scoring. Second, the representation diagnostics and PCI-based group separation results indicate that augmented descriptions and multi-level structural scoring provide meaningful discrimination between contrasting academic profiles. Third, the ablation, ranking, and traceability analyses show that the framework components contribute complementary effects rather than uniformly improving every separation metric, and the final ranking outputs can be traced back to course-level semantic evidence.

5.8. Sensitivity and Ranking Stability Analysis

To assess whether the rankings produced by the proposed framework depend on specific design choices, we conducted a one-factor-at-a-time (OFAT) sensitivity analysis. Starting from the original configuration as the baseline (B0)—Top-1 path locking, equal

hierarchical weighting in the PCI, grade-weighted NTACS scoring, and credit-weighted transcript evidence—we varied a single assumption at a time and measured how far the resulting student ranking departed from B0. Agreement with the baseline ranking was quantified using Spearman’s rank correlation (ρ), which captures changes in the overall ordering, and Top-10 overlap, which captures whether the highest-ranked candidates remain in the same group. The analysis was performed independently for all four evaluated target paths/domains (Data Scientist, Data Analyst, Data Engineer, and Visual Art), consistent with the ranking results reported in Table 18.

This analysis evaluates ranking stability under the five one-factor-at-a-time assumption changes (SA1–SA5) tested in this study. Therefore, it is distinct from and complementary of the ablation study (Section 5.6), the cross-method comparison (Section 5.7), and the expert-labelled validation, which assess the framework’s comparative effectiveness and discriminative value rather than its stability. To avoid confusion with the reference methods (M1–M5), sensitivity-analysis configurations are denoted SA1–SA5. SA1 (grade-free ranking) quantifies the influence of grade evidence by replacing NTACS with a grade-free relevant credit volume (RCV) score. SA2 (Top-3 path averaging) tests sensitivity to the L_3 entry-point selection rule and sensitivity to candidate-path ambiguity. SA3 (leaf-weighted PCI) and SA4 (job-weighted PCI) assess dependence on the equal hierarchical-level weighting assumption in the PCI by emphasising the leaf level (L_3) and job-category level (L_1), respectively. SA5 (equal course weighting) tests whether course-credit magnitudes drive the results by replacing actual credits with an unweighted-course assumption. Table 22 lists each scenario with the single assumption modified relative to B0.

Table 22. Sensitivity scenarios and the modified assumption relative to the baseline (B0). Each scenario modifies exactly one assumption; all other components are held fixed.

ID	Factor	Change from B0
SA1	Grade evidence	Compare grade-weighted NTACS with the grade-free IF-weighted relevant credit volume (RCV, Equation (17)).
SA2	L_3 entry selection	Replace Top-1 path locking with Top-3 candidate-path averaging.
SA3	PCI level weighting	Re-weight PCI levels with $(w_1, w_2, w_3) = (0.25, 0.25, 0.50)$, emphasising the leaf level (L_3).
SA4	PCI level weighting	Re-weight PCI levels with $(w_1, w_2, w_3) = (0.50, 0.25, 0.25)$, emphasising the job-category level (L_1).
SA5	Credit weighting	Replace actual course credits with an unweighted-course assumption.

Across all scenarios, only one factor is altered at a time, and all remaining components are held fixed to the baseline, ensuring that any observed change in ranking can be attributed to the tested assumption.

5.8.1. Sensitivity Analysis Results

Agreement with the baseline ranking is summarised in Tables 23 and 24, with Table 23 reporting Spearman’s rank correlation (ρ) and Table 24 reporting the corresponding Top-10 overlap. For the parameter-perturbation scenarios (SA2, SA3, and SA4) and the credit scenario (SA5), high agreement indicates that the ranking is stable under the tested change. Scenario SA1 is interpretive in a different sense: because RCV is a grade-free construct rather than a perturbed version of NTACS, low or even negative agreement is expected by design and should be interpreted as evidence of grade influence rather than as ranking instability.

Table 23. Ranking agreement with the baseline (B0) under each sensitivity scenario, reported as Spearman’s rank correlation (ρ ; higher indicates greater stability).

Scenario	Data Scientist	Data Analyst	Data Engineer	Visual Art
B0 (reference)	1.000	1.000	1.000	1.000
SA2: Top-3 path averaging	0.967	0.939	0.963	0.990
SA3: Leaf-weighted PCI	0.999	0.999	0.999	0.999
SA4: Job-weighted PCI	0.999	0.999	0.999	0.999
SA5: Equal-course weighting	0.991	0.976	0.990	0.988
SA1: Grade-free ranking	−0.101	−0.226	−0.096	0.409

Table 24. Ranking agreement with the baseline (B0) under each sensitivity scenario, reported as Top-10 overlap with the baseline ranking (higher indicates greater stability among top candidates).

Scenario	Data Scientist	Data Analyst	Data Engineer	Visual Art
B0 (reference)	1.000	1.000	1.000	1.000
SA2: Top-3 path averaging	0.700	0.700	0.500	0.900
SA3: Leaf-weighted PCI	1.000	1.000	1.000	1.000
SA4: Job-weighted PCI	0.900	1.000	1.000	1.000
SA5: Equal course weighting	0.800	0.800	0.900	1.000
SA1: Grade-free ranking	0.200	0.100	0.300	0.400

To improve readability, we report ranking stability using two complementary agreement measures. Spearman’s rank correlation (ρ) captures changes in the overall ordering of candidates, while Top-10 overlap captures whether the highest-ranked candidates remain largely unchanged. Table 23 reports ρ values for each scenario and target path, and Table 24 reports the corresponding Top-10 overlap values.

In both tables, higher values indicate greater stability for the parameter-perturbation scenarios (SA2, SA3, SA4, and SA5), so low values are expected by design.

While ρ summarises global stability across the full ranking list, Top-10 overlap focuses on stability at the decision-critical top end of the list. We therefore report Top-10 overlap separately in Table 24.

Together, these two views distinguish scenarios that preserve the overall ordering from those that mainly reshuffle the top-ranked candidates, which is particularly relevant for shortlisting-based recruitment decisions.

5.8.2. Interpretation of Sensitivity and Ranking Stability

The ranking is highly stable with respect to hierarchical-level weighting. Re-weighting the three PCI levels towards either the leaf level (SA3) or the job-category level (SA4) leaves the ordering almost unchanged ($\rho \geq 0.998$ across all target paths/domains, with Top-10 overlap of 0.90–1.00). This indicates that the rankings do not depend strongly on the specific choice of equal level weighting in the PCI within the tested weighting alternatives. Stability is also high for the credit assumption (SA5): replacing actual course credits with an unweighted-course assumption preserves the ordering closely ($\rho = 0.976$ – 0.991), showing that the rankings are not driven primarily by course-credit magnitudes. For the L_3 entry-point rule (SA2), the overall ordering remains stable ($\rho = 0.939$ – 0.990), although the Top-10 overlap is more variable (0.50–0.90). This indicates that averaging over the Top-3 candidate paths preserves the global ranking while modestly reshuffling the top of the list, suggesting that single-best (Top-1) path locking mainly affects the composition of the highest-ranked candidates without substantially altering the broader ranking structure.

Scenario SA1 compares the grade-weighted ranking (NTACS) with the grade-free relevant-credit-volume ranking (RCV). For the three data-domain roles, the two rankings

are essentially uncorrelated or weakly negatively correlated (ρ between -0.226 and -0.096), and the Top-10 overlap is small (0.10 – 0.30). This is expected and informative rather than a sign of instability: NTACS and RCV measure different constructs. NTACS captures the quality of learning, represented by grades and weighted by relevance, whereas RCV captures the volume of relevant coursework, irrespective of grades. A student who has taken many job-relevant courses (high RCV) need not be the same student who achieved the highest grades in relevant courses (high NTACS), so the two orderings can diverge substantially. The result confirms that grade evidence is a decisive factor in the NTACS ranking by design, which is why the framework also reports RCV (Equation (17)) and the relevant-volume measure (Equation (14)) as complementary, grade-independent indicators. For the Visual Art path, the divergence is weaker ($\rho = 0.409$), indicating closer agreement between grade-weighted and grade-free evidence than in the data-domain roles. This may suggest that, in this path, the volume of relevant coursework and grade-weighted achievement are more closely aligned.

Across the parameter-perturbation scenarios, the proposed ranking is generally stable, with Spearman correlations ranging from 0.939 to 0.999 . This indicates that the results are not an artefact of a single arbitrary assumption regarding hierarchical weighting, the L_3 entry-point rule, or course credits. The only factor that materially changes the ordering is the inclusion of grade evidence, which reflects the intended grade-sensitive design of NTACS and motivates the complementary grade-free measures retained in the framework.

5.9. Evaluation of Ranking Validity and Expert Agreement

This subsection evaluates the ranking validity of the proposed framework and the reliability of the expert-based reference judgement. The evaluation is organised into four parts: construction of the expert-based reference set, grade-free relevance-ranking performance, grade-effect diagnostics, and expert-rating reliability and system–expert agreement.

Because employer hiring decisions, placement outcomes, and recruiter shortlisting records were not available for the datasets used in this study, an expert-labelled reference was constructed. This reference is not treated as an error-free ground truth. Rather, it serves as an independent expert judgement for assessing whether the rankings produced by the framework are reasonably aligned with human assessment of student–job fit.

It is important to note that this evaluation involves an expert activity separate from the HITL ontology verification described in Section 4.4. The HITL stage used two panels totalling ten members to validate LLM-proposed ontology mappings—a knowledge-engineering task. The ranking-validation stage described here used a distinct panel of five evaluators to rate student–job suitability from transcript evidence—a competency-assessment task. The two panels serve different functions and were selected independently on the basis of the expertise required for each task.

5.9.1. Construction of the Expert-Based Reference Set

The original ranking experiment included 94 students: 73 computing students (Computer Engineering: 20; Business Information Systems: 21; Computer Science: 21; Information Technology: 11) and 21 Visual Arts students. Because programme sizes were unequal, proportional sampling would have resulted in very few Information Technology students being included. To ensure sufficient representation across all strata and to prevent expert survey fatigue from rating all 94 candidates, disproportionate stratified sampling was applied: exactly nine students were drawn from each of the five programmes (Computer Engineering, nine students; Business Information Systems, nine students; Computer Science, nine students; Information Technology, nine students; Visual Arts, nine students), producing a balanced expert-evaluation subset of 45 students. Each sampled student was

matched to the full ranking pool by student ID so that framework-generated scores for the 45 candidates could be directly compared with expert ratings.

Four job postings were selected for expert evaluation—one per target job type—based on the spread of NTACS scores across the 45 sampled students so that postings producing meaningful variation in student–job fit were preferred: task A1 used posting P1 (Senior Data Scientist), task A2 used posting P2 (Analyst, Imagery Analytics), task A3 used posting P3 (Principal Data Engineer), and task A4 used posting P4 (Art & Graphic Design Team Leader). Each task generated 225 ratings, yielding 900 rating records in total.

The expert panel comprised five evaluators: three HR professionals from private-sector companies with more than ten years of experience in recruitment and competency assessment and two university career-guidance lecturers with relevant experience in student employability. Each expert rated the suitability of each of the 45 candidates for each selected job posting on a 1–5 scale.

5.9.2. Relevance-Ranking Performance

Table 25 reports the grade-free relevance-ranking performance of four methods: keyword matching (M2), flat SBERT matching (M3), L_3 -only ontology matching with the importance factor (M4), and the grade-free variant of the proposed framework (M5: NonGrade). The GPA baseline (M1) and the full NTACS score (M5: Full Framework) are excluded from this table because they incorporate academic grade information and therefore do not represent pure relevance scores.

Table 25. Relevance-ranking performance of grade-free methods (M2–M5: NonGrade). M1 and M5 (Full Framework) are excluded because they include grade information.

Job Type	Method	AUC	Cliff's δ	Recall@20	HitRate@20	Mean Rank
Data Scientist	M2: Keyword Matching	1.0000	1.0000	0.2740	1.00	37.0000
Data Scientist	M3: Flat SBERT	1.0000	1.0000	0.2740	1.00	37.0000
Data Scientist	M4: L_3 -Only + IF	0.5740	0.1481	0.2603	0.95	45.9452
Data Scientist	M5: NonGrade	0.4442	−0.1115	0.2192	0.80	48.6712
Data Analyst	M2: Keyword Matching	0.7776	0.5551	0.2192	0.80	41.6712
Data Analyst	M3: Flat SBERT	1.0000	1.0000	0.2740	1.00	37.0000
Data Analyst	M4: L_3 -Only + IF	0.9583	0.9165	0.2740	1.00	37.8767
Data Analyst	M5: NonGrade	0.9667	0.9335	0.2740	1.00	37.6986
Data Engineer	M2: Keyword Matching	1.0000	1.0000	0.2740	1.00	37.0000
Data Engineer	M3: Flat SBERT	1.0000	1.0000	0.2740	1.00	37.0000
Data Engineer	M4: L_3 -Only + IF	1.0000	1.0000	0.2740	1.00	37.0000
Data Engineer	M5: NonGrade	1.0000	1.0000	0.2740	1.00	37.0000
Visual Art	M2: Keyword Matching	0.8826	0.7652	0.6190	0.65	19.5714
Visual Art	M3: Flat SBERT	1.0000	1.0000	0.9524	1.00	11.0000
Visual Art	M4: L_3 -Only + IF	1.0000	1.0000	0.9524	1.00	11.0000
Visual Art	M5: NonGrade	1.0000	1.0000	0.9524	1.00	11.0000

Five statistics are reported. AUC measures the probability that a randomly selected target-group student receives a higher score than a randomly selected reference-group student. Cliff's δ provides a non-parametric effect size for group separation. Recall@20 measures the proportion of target-group students retrieved within the Top-20 ranked students; HitRate@20 measures the proportion of Top-20 positions occupied by target-group students; and Mean Rank is the mean-rank position of target-group students. For data-oriented job types, computing students are the target group, and Visual Arts students are the reference group; for the Visual Art task, the roles are reversed.

The results show strong grade-free separation for several job types. For Data Engineer, all four grade-free methods achieved perfect separation. For Data Analyst, M5 (NonGrade) produced an AUC of 0.9667 and Cliff's δ of 0.9335—slightly higher than M4, which suggests

that the ontology-based relevance structure is useful for this job type. For the Visual Art task, M3, M4, and M5 (NonGrade) all achieved $AUC = 1.0000$, Cliff's $\delta = 1.0000$, and $HitRate@20 = 1.00$.

The Data Scientist case warrants more cautious interpretation. Although M2 and M3 achieved perfect separation, M4 and M5 (NonGrade) produced weaker results, with M5 (NonGrade) yielding $AUC = 0.4442$ and Cliff's $\delta = -0.1115$. Diagnostic inspection indicated that some data-science skill assignments were unexpectedly associated with the Mechanical L_3 node, suggesting a semantic nearest-node assignment issue rather than a failure of the weighting mechanism itself. This case illustrates that O4CM is not presented as error-free; rather, its ontology-based traceability makes failure cases diagnosable—the semantic traceability map surfaces the specific L_3 node responsible for the misalignment, enabling targeted correction of the augmentation or path-locking step.

It should also be noted that the maximum achievable $Recall@20$ depends on the target group size. For the 73 computing students, the maximum is $20/73 = 0.2740$; for the 21 Visual Arts students, it is $20/21 = 0.9524$. Therefore, $Recall@20$ should be interpreted alongside $HitRate@20$ and Mean Rank.

5.9.3. Grade-Effect Diagnostics

Table 26 examines how including grades changes the student ranking. M5 (NonGrade) is the grade-free ontology-based relevance score, while M5 (Full Framework) is the full NTACS score that incorporates academic grades. The Spearman correlation between M5 (Full Framework) and M5 (NonGrade) indicates whether grades preserve the relevance-based ordering; the correlation between GPA and M5 (Full Framework) indicates how strongly the full score is driven by general academic performance.

Table 26. Grade-effect diagnostics comparing M5 (Full Framework) with M5 (NonGrade).

Job Type	$\rho(\text{Full,NonGrade})$	$\rho(\text{GPA,Full})$	$r(\text{GPA,Full})$	Median Shift	Mean Abs. Shift	Median Abs. Shift
Data Scientist	0.2105	0.9075	0.9072	+1.75	26.6064	21.50
Data Analyst	−0.1239	0.8029	0.8149	−5.75	34.1170	34.25
Data Engineer	−0.1765	0.9388	0.9355	+2.25	34.3191	31.00
Visual Art	0.4295	0.9805	0.9838	−2.00	22.6277	17.00

M5 (Full Framework) is strongly associated with GPA across all job types (Spearman $\rho = 0.80$ – 0.98), which is expected because grades appear directly in the NTACS numerator. In contrast, the correlation between M5 (Full Framework) and M5 (NonGrade) is weak or negative for the three data-domain roles ($\rho = -0.18$ to 0.21), indicating that incorporating grades substantially changes the relevance-based ranking. Shifts in mean absolute rank ranged from 22.6 to 34.3 positions, confirming that many students change positions after grades are added. Therefore, M5 (Full Framework) should be interpreted as a grade-sensitive competency-quality score rather than a pure relevance score; M5 (NonGrade) is more appropriate for evaluating grade-free ontology-based relevance.

5.9.4. Expert-Rating Reliability

Before comparing system-generated scores with expert ratings, the reliability of the expert judgements was examined. Each of the four tasks involved 45 candidates rated by 5 experts, giving 225 ratings per task and 900 ratings in total. Table 27 reports ICC(2,k) under a two-way random-effects absolute-agreement model, Krippendorff's α , and Kendall's W for each posting and overall.

ICC(2,k) values ranged from 0.77 to 0.79 across all four postings, indicating that the averaged expert rating is sufficiently reliable for use as an aggregated reference. However,

Krippendorff's α (0.39–0.42) reflects only low-to-moderate agreement in exact score assignment among individual raters, and Kendall's W (0.43–0.55) similarly indicates moderate ranking concordance. These values indicate that while the aggregated mean rating is stable, individual expert scores varied, which is consistent with the subjective nature of competency assessment across different professional backgrounds. Therefore, the mean expert rating is used as an aggregated independent reference, not as an error-free ground-truth label.

Table 27. Expert-rating reliability across four job postings.

Task	Posting	Job Type	ICC(2,k)	Kripp. α	Kendall's W
A1	P1	Data Scientist	0.7857	0.4126	0.4892
A2	P2	Data Analyst	0.7939	0.4167	0.5511
A3	P3	Data Engineer	0.7700	0.3852	0.5276
A4	P4	Visual Art	0.7711	0.3857	0.4305

5.9.5. External Validation Against Expert Ratings

After confirming the reliability of the aggregated expert ratings, each ranking method was compared with the mean expert ratings for the 45 students in the expert-evaluation subset. The merged validation dataset contained 180 candidate–posting pairs (45 candidates $\times$ 4 postings) with no missing system scores.

Rank-based metrics were used: Spearman's ρ and Kendall's τ measure overall rank agreement between system scores and expert mean ratings; NDCG@10 and NDCG@20 measure whether candidates with high expert ratings are placed near the top of the system-generated ranking.

The external validation results for all ranking methods across the four job types are summarised in Table 28.

Table 28. External validation against mean expert ratings for all methods across four job types. Bold values indicate the highest-performing method for each metric within each job type.

Job Type	Method	Spearman's ρ	Kendall's τ	NDCG@10	NDCG@20
Data Scientist	M1: GPA Ranking	0.6592	0.4885	0.8676	0.8782
Data Scientist	M2: Keyword Matching	0.1531	0.1155	0.6729	0.7237
Data Scientist	M3: Flat SBERT	−0.0257	−0.0169	0.5423	0.6362
Data Scientist	M4: L_3 -Only + IF	0.5266	0.3891	0.8183	0.8586
Data Scientist	M5: NonGrade	0.4371	0.3274	0.7885	0.8506
Data Scientist	M5: Full Framework	0.5534	0.4102	0.6830	0.7987
Data Analyst	M1: GPA Ranking	0.6585	0.5121	0.9265	0.9190
Data Analyst	M2: Keyword Matching	0.0608	0.0400	0.6933	0.7606
Data Analyst	M3: Flat SBERT	0.0191	0.0273	0.5545	0.6634
Data Analyst	M4: L_3 -Only + IF	0.1310	0.0850	0.5711	0.7092
Data Analyst	M5: NonGrade	0.1103	0.0810	0.5632	0.6868
Data Analyst	M5: Full Framework	0.6500	0.4859	0.9290	0.9190
Data Engineer	M1: GPA Ranking	0.7267	0.5503	0.9011	0.9101
Data Engineer	M2: Keyword Matching	−0.0610	−0.0223	0.6090	0.6622
Data Engineer	M3: Flat SBERT	−0.1262	−0.0801	0.5177	0.6283
Data Engineer	M4: L_3 -Only + IF	0.4382	0.2963	0.7472	0.7979
Data Engineer	M5: NonGrade	0.4012	0.2659	0.7119	0.7767
Data Engineer	M5: Full Framework	0.7082	0.5406	0.9014	0.9040
Visual Art	M1: GPA Ranking	0.6946	0.5106	0.7379	0.8432
Visual Art	M2: Keyword Matching	0.5768	0.4362	0.7951	0.8607
Visual Art	M3: Flat SBERT	0.6401	0.4903	0.9878	0.9387
Visual Art	M4: L_3 -Only + IF	0.2670	0.1237	0.9040	0.8497
Visual Art	M5: NonGrade	0.2696	0.1375	0.9327	0.8711
Visual Art	M5: Full Framework	0.7251	0.5356	0.7942	0.8710

The results show that expert ratings were more strongly aligned with grade-sensitive scores than with grade-free relevance scores in several job types. For Data Scientist, GPA ranking produced the highest Spearman correlation ($\rho = 0.6592$), followed by M5 (Full Framework) ($\rho = 0.5534$) and M4 ($\rho = 0.5266$). For Data Analyst, M1 and M5 (Full Framework) showed very close rank agreement with expert ratings ($\rho = 0.6585$ and 0.6500 respectively), and M5 (Full Framework) produced the highest NDCG@10. For Data Engineer, M1, again, achieved the highest Spearman correlation ($\rho = 0.7267$), closely followed by M5 (Full Framework) ($\rho = 0.7082$). For Visual Art, M5 (Full Framework) achieved the highest overall rank agreement ($\rho = 0.7251$), while M3 produced the highest NDCG@10 values.

Overall, these results indicate that expert judgements were not based solely on transcript–job relevance. Experts also appear to have considered academic achievement, which explains the alignment between GPA-sensitive scores and expert ratings. M5 (Non-Grade) remains the most appropriate measure for evaluating grade-free ontology-based relevance, while M5 (Full Framework) provides a grade-sensitive competency-quality score that is more consistent with expert judgement in several scenarios. At the same time, these analyses show that PCI, TACS, and NTACS provide complementary and traceable competency-based evidence beyond GPA, keyword, and flat-SBERT baselines rather than universally outperforming all simpler methods.

5.10. Expert-Aligned Ranking Within Computing Programmes

The full-cohort evaluation in Section 5.9 includes both computing-programme students and Visual Arts students, which creates a broad programme-level contrast that may inflate separation scores. To examine whether the framework retains meaningful agreement with expert judgements under more demanding conditions, we conducted a computing-only expert-aligned ranking analysis in which Visual Arts candidates and Visual Arts postings were excluded.

5.10.1. Experimental Setting

The analysis retained only expert-rated candidates from Computer Engineering (CE), Business Information Systems (BIS), Computer Science (CS), and Information Technology (IT), spanning four closely related computing programmes. Only three data-domain job postings were included: Data Scientist, Data Analyst, and Data Engineer. This resulted in a restricted evaluation setting involving 36 computing candidates and 108 candidate–job pairs, each evaluated against mean expert ratings. Rankings were compared against expert judgements using Spearman’s ρ , Kendall’s τ , NDCG@10, and NDCG@20. M1 (GPA Ranking) was excluded from this analysis because the primary question is whether ontology-based relevance signals can distinguish among students from closely related disciplines.

5.10.2. Expert-Aligned Ranking Results

Table 29 reports the system–expert agreement for each method across the three data-domain job types.

5.10.3. Interpretation of Computing-Only Expert Alignment

Across all three data-domain job types, M5 (Full Framework) achieved the strongest agreement with expert judgements ($\rho = 0.81, 0.70$, and 0.72 for Data Scientist, Data Analyst, and Data Engineer, respectively; all $p < 0.001$). This pattern is consistent with the full-cohort results and indicates that incorporating academic performance into ontology-weighted scoring produces rankings that more closely reflect expert assessments, even when the evaluation is restricted to computing-oriented programmes with partial curricular overlap.

Table 29. System–expert agreement in the computing-only evaluation (36 candidates; CE, BIS, CS, and IT programmes; three data-domain job types). The highest value per metric per job type is bolded. M1 is excluded, as the question concerns ontology-based discrimination within computing disciplines.

Job Type	Method	Spearman's ρ	Kendall's τ	NDCG@10	NDCG@20
Data Scientist	M2: Keyword Matching	−0.1415	−0.1097	0.6729	0.7249
Data Scientist	M3: Flat SBERT	−0.4728	−0.3262	0.5423	0.6373
Data Scientist	M4: L_3 -Only + IF	0.5090	0.3727	0.8221	0.8573
Data Scientist	M5: NonGrade	0.4965	0.3623	0.8448	0.8614
Data Scientist	M5: Full Framework	0.8100	0.6354	0.8881	0.9325
Data Analyst	M2: Keyword Matching	−0.2349	−0.1717	0.7072	0.7783
Data Analyst	M3: Flat SBERT	−0.4277	−0.2897	0.5545	0.6634
Data Analyst	M4: L_3 -Only + IF	−0.2365	−0.1814	0.5716	0.6946
Data Analyst	M5: NonGrade	−0.2394	−0.1811	0.5640	0.6896
Data Analyst	M5: Full Framework	0.7006	0.5409	0.9131	0.9311
Data Engineer	M2: Keyword Matching	−0.3814	−0.2720	0.6090	0.6635
Data Engineer	M3: Flat SBERT	−0.4996	−0.3605	0.5177	0.6295
Data Engineer	M4: L_3 -Only + IF	0.1482	0.0987	0.6769	0.7567
Data Engineer	M5: NonGrade	0.0695	0.0383	0.5813	0.7300
Data Engineer	M5: Full Framework	0.7207	0.5296	0.8947	0.9046

Grade-free methods (M3, M4, and M5) produced weakly negative or near-zero Spearman correlations for Data Analyst and Data Engineer, suggesting that ontology-based relevance signals alone are insufficient for fine-grained discrimination among students from closely related disciplines when grades are excluded. The negative correlations for M3 across all three job types further indicate that flat semantic matching without hierarchical structural weighting is not robust under this more demanding evaluation condition.

These findings should be interpreted cautiously. The expert subset comprises 36 candidates drawn from a single institution, and caution is warranted in generalising to broader settings. Accordingly, the computing-only analysis is offered as supporting evidence rather than definitive proof of generalisability. Nevertheless, the results suggest that the framework's performance in the full-cohort evaluation cannot be attributed solely to obvious programme-level contrasts: M5 (Full Framework) retains substantial agreement with expert judgements when evaluated exclusively within closely related computing programmes and data-domain occupations.

6. Discussion

6.1. Interpretation of Key Findings

The findings provide empirical support for O4CM as a prototype framework for ontology-grounded student-to-job competency mapping within the dataset used in this study. Rather than demonstrating universal superiority over all possible matching systems, the results show that the proposed framework offers a coherent and traceable way to integrate academic evidence, ontology-based structural alignment, and job-side relevance signals into a single ranking pipeline.

First, the ontology induction results support the contribution of a semi-automated ontology construction process. The results show that LLM-assisted construction can produce a usable ontology backbone when its outputs are treated as candidate structures rather than final assertions. The five-model ensemble produced high observed agreement for the knowledge-to-skill mapping task, with 21 of 22 knowledge domains reaching at least 4/5 consensus. However, the expert correction cases also show that model agreement is not equivalent to domain correctness. The unanimous case of Administration and Management, which was reassigned from Cognitive_Skills to Technical_Skills after expert

review, is particularly important. It indicates that even highly consistent LLM outputs may still be functionally misaligned with the target professional context. This supports the methodological decision to apply HITL verification across all agreement tiers, including unanimous outputs.

Second, the semantic representation diagnostics and PCI-based group separation results support the contribution of multi-level semantic and structural discrimination. The diagnostic examples show why surface labels alone are insufficient: short course titles, skill names, and ontology labels often omit the functional context needed for semantic comparison. The PCI-based group separation analysis further shows that the locked-path structural score captures meaningful curriculum-level differences between the two clearly contrasting academic profiles that were evaluated. Computing students obtain higher credit-weighted PCI scores on the Computer Job path, whereas Visual Arts students obtain higher scores on the Art Job path, with statistically significant and large effects in both directions. This bidirectional pattern supports the internal structural validity of the ontology backbone and indicates that the PCI functions as more than a leaf-level similarity score.

Third, the ablation and ranking results support the contribution of a job-conditioned and traceable scoring framework. The proposed framework does not maximise every individual metric. This is expected because the goal is not to optimise separation on a single path but to maintain balanced behaviour across job-conditioned ranking, negative-control cross-domain discrimination, and traceability. Removing $IF_{c,k}$ weakens Computer Job separation, indicating that job-side structural relevance is important in differentiating data-domain competency evidence. Removing the PCI affects the Art Job path more strongly, suggesting that multi-level structural scoring helps reduce cross-domain contamination in the evaluated computing-versus-arts negative-control setting. Removing augmentation improves some Computer Job separation values but weakens Art Job separation, implying that short labels may over-specialise one domain while reducing robustness across contrasting paths.

6.2. Role of the Framework Components

The results clarify the distinct function of each major component in O4CM. Semantic augmentation provides richer contextual evidence for embedding and supports interpretability at the course and ontology-node levels. It is especially useful when course titles or skill labels are too short to express the underlying competency. However, the ablation results also show that augmentation should not be interpreted as a mechanism that always increases numerical separation. Its main value is to provide richer and more balanced semantic evidence, not to maximise one separation metric.

The PCI mechanism provides structural verification across the ontology path. A course may be close to a knowledge node at L_3 , but this does not guarantee that it is also consistent with the corresponding skill group at L_2 or job category at L_1 . By averaging the level-wise similarity scores across the locked path, PCI_i provides a continuous structural confidence weight. This design avoids hard exclusion of ambiguous courses while reducing their contribution when higher-level contextual support is weak.

The job importance factor ($IF_{c,k}$) plays a different role. In the current formulation, it represents the mean job-side structural confidence of job-skill records mapped to knowledge node k for job type c . It should therefore be interpreted as job-side structural relevance, not the raw frequency of occurrence. This distinction is important because the framework does not simply reward commonly appearing skill terms. Instead, it rewards knowledge nodes whose job-side skill evidence is structurally supported by the ontology path.

Together, PCI_i and $IF_{c,k}$ create a two-sided relevance mechanism. The former evaluates how strongly a student's course aligns with the ontology path, while the latter evaluates

how strongly that knowledge node is supported by job-side evidence for a target job type. This interaction is the main reason why NTACS_c is more informative than GPA, keyword overlap, or flat similarity alone.

6.3. Comparison with Simpler Matching Strategies

The comparison methods help position O4CM relative to common ranking and matching strategies. GPA ranking provides a useful lower-bound reference because it reflects academic performance without considering job relevance. In this dataset, the Visual Arts control group has the highest mean GPA. As a result, GPA-based ranking fails the expected direction for data-domain roles. This finding supports the argument that GPA alone is not sufficient for job-specific competency assessment because it cannot distinguish whether strong academic performance was achieved in job-relevant or job-irrelevant coursework.

Keyword matching performs better than GPA for broad domain separation, but its identical separation values across Data Scientist, Data Analyst, and Data Engineer indicate limited sub-type sensitivity. This limitation is consistent with the nature of lexical matching: roles within the same broad domain often share surface vocabulary, even when they differ in competency emphasis. Therefore, keyword matching can identify general data-domain relevance but is less suitable for distinguishing closely related occupational sub-types.

Flat SBERT similarity improves over purely lexical matching by using dense semantic representations [3]. However, the results show that semantic encoding alone is not sufficient to satisfy all evaluation criteria for ontology-grounded competency mapping. Without branch locking and multi-level PCI scoring, the method lacks an explicit mechanism for checking whether a leaf-level match is also consistent with the broader skill and job-category context. O4CM retains the benefit of SBERT-based semantic similarity while adding ontology-based structural verification and job-side relevance weighting.

The L_3 -only comparator further clarifies the contribution of multi-level scoring. Because it uses the top-matched knowledge node and retains $\text{IF}_{c,k}$, it is stronger than flat SBERT. However, its weaker Visual Art performance suggests that leaf-level matching can still allow for cross-domain contamination when higher-level ontology support is not considered. The proposed framework addresses this by using the full locked path from L_3 to L_2 and L_1 .

Taken together, the comparison results show that no single simpler alternative satisfies all four evaluation criteria simultaneously. GPA ranking fails in domain discrimination for data roles. Keyword matching and flat SBERT achieve broad separation but cannot distinguish sub-types (Data Scientist vs. Data Analyst vs. Data Engineer). The L_3 -only method with $\text{IF}_{c,k}$ achieves sub-type sensitivity but degrades on the cross-domain path. Only the proposed framework satisfies domain discrimination, sub-type sensitivity, cross-domain robustness, and ontology-based interpretability in this evaluation. The external validation results (Section 5.9.5) further show that M5 (Full Framework) aligns more closely with expert judgement than keyword or flat SBERT methods in most job types, while M5 (NonGrade) provides a grade-free relevance view that is not available from simpler baselines.

One limitation of this evaluation is that it was conducted primarily on a computing-arts contrast, which is a relatively easy discrimination task. The framework has not yet been tested on near-boundary curricula such as Data Science, Software Engineering, Business Analytics, and Human-Computer Interaction, where job-relevant knowledge domains overlap more substantially and the expected benefit of multi-level structural scoring may be less pronounced. Near-boundary testing is an important direction for future work and would provide a more stringent practical comparison with simpler alternatives.

6.4. Application-Oriented Significance

From an application perspective, the main value of O4CM is not only the final rank order but also the structured evidence attached to each ranking decision. The framework operates at the course-record level, which allows it to capture variation within the same academic programme. This is important for programmes with flexible elective structures, where two students may share the same major but accumulate different job-relevant competency profiles.

The normalised score ($NTACS_c$) is useful for candidate ranking because it reduces sensitivity to transcript length and credit volume while preserving the effects of grades, credits, student-side structural alignment, and job-side relevance. The supplementary $RelVol_c$ score provides an additional view of the breadth of structurally supported coursework, independent of grade performance. These two outputs can support different decision needs: $NTACS_c$ for ranking and $RelVol_c$ for understanding the amount of relevant learning evidence behind the score.

The Semantic Traceability Map extends the framework beyond black-box ranking. By aggregating the course-level relevance signal (E_i) across knowledge nodes or skill groups, the framework can show which parts of a student's transcript contribute most strongly to a target job type. This supports practical use cases in educational advising, curriculum review, career guidance, and recruitment screening. For example, the framework can identify whether a student is strong because of technical computing courses, cognitively oriented analytical courses, or broader interdisciplinary coursework. Such evidence is difficult to obtain from a single GPA score or an unstructured semantic similarity score.

6.5. Implications for Ontology-Grounded Competency Mapping

The study suggests that ontology-grounded matching is most useful when it is treated as an auditable analytical framework rather than a fully automatic truth-generating system. The ontology backbone in O4CM is constructed from O*NET knowledge domains, LLM-induced skill groups, and expert-confirmed job-category assignments. This design provides a controlled semantic structure for the mapping of student and job evidence while still acknowledging that the resulting hierarchy is an analytical taxonomy rather than a complete representation of all possible competency relationships.

The HITL results also show that expert review is not merely a quality assurance step applied after automation. It is a necessary part of the ontology construction process. The LLM ensemble helps reduce manual workload by producing candidate structures and highlighting agreement patterns, but expert judgement is required to determine whether those structures are appropriate for the target domain. This balance is important for framework-oriented applications, where scalability and auditability must both be maintained.

6.6. Limitations

Several limitations qualify the scope of the present findings, and each is described here in terms of how it may shape the observed results rather than merely listed.

First, the empirical evaluation is internal to a single Thai university. Because the transcript records originate from one institution, the observed score distributions are partly a product of local curriculum design, grading practices, course-naming conventions, and programme structure. These institution-specific factors enter the pipeline directly: course titles and descriptions drive the SBERT augmentation in Phase 3, while grades and credits enter the $NTACS$ numerator in Phase 5. Consequently, the absolute score ranges reported here and, to some extent, the magnitude of the group separations may shift if the framework is applied to an institution with different credit systems, grading scales, or curricular

vocabulary. Therefore, the results should be read as evidence of internal feasibility rather than as a generalisation claim across institutions or national education systems.

Second, the composition of the evaluated cohort influences the observed separation. The four computing programmes were selected as the positive group, while Visual Arts was selected specifically because it is academically distant from data-domain requirements. This positive-versus-control design supports a clear negative-control test, but it does not, on its own, demonstrate performance in programmes with intermediate or partial overlap. The effect of cohort composition is compounded by a grading confound: the B.F.A. Visual Arts group has the highest mean GPA in the sample (Table 7). Because GPA enters every grade-sensitive score, this distribution works against the Visual Arts group under NTACS and GPA ranking in the data-domain direction while favouring it on the Art Job path. This is precisely why the framework also reports grade-free measures (RCV_c and the relevant-volume measure) as a GPA-independent view of competency evidence and the grade-sensitive and grade-free results should be weighed together rather than in isolation.

Third—and most consequential for interpreting the validation—the computing-versus-arts comparison is a relatively easy contrast. It establishes that O4CM can separate clearly different academic profiles, but this is not equivalent to demonstrating that the scoring mechanism is robust for difficult, near-boundary competency-mapping cases. The computing-only analysis in Section 5.10 begins to probe this harder setting and is informative about the framework’s discriminative ceiling. There, Information Technology was separated consistently from the core computing programmes with medium-to-large effects (mean $|d| = 1.06$), whereas Computer Engineering, Business Information Systems, and Computer Science remained closely clustered (mean $|d| = 0.24$ for the CE/BIS/CS pairs). This pattern is consistent with established curricular analysis: the ACM/IEEE-CS Computing Curricula 2020 report documents substantial shared knowledge across the core computing disciplines, as evidenced by its cross-disciplinary mapping of computing knowledge areas, while characterising information technology as the discipline concerned most directly with concrete technology components in organisational settings [36]. The clustering of the computing-heavy Business Information Systems programme with Computer Science and Computer Engineering rather than with Information Technology is consistent with the substantial data-domain course content in its curriculum (Table 6). As noted in Section 5.10.3, however, the present data cannot fully distinguish whether the low within-core separation reflects genuine curricular overlap or a limit of the ontology’s discriminative granularity, and near-boundary programmes such as Information Systems, Business Analytics, Software Engineering, Digital Media, Human–Computer Interaction, Computational Design, and Applied Statistics would be required to resolve this question. Their absence here bounds the strength of the validation. A further contributor at the dataset level is the degree of enrolment overlap among the core computing programmes: students from CE, BIS, and CS in this cohort share substantial course content, with programmes differing mainly in course naming conventions rather than in the underlying competency coverage reflected in transcripts. This structural similarity makes it difficult to demonstrate clear programme-level discrimination experimentally within the current dataset and may understate the framework’s potential precision when applied to genuinely distinct curricula in future evaluations.

Fourth, the job-side evidence is derived from a static and heterogeneous job-posting corpus, and this affects the importance factor ($IF_{c,k}$) that conditions every ranking. The data-domain postings originate from an Indeed-sourced Kaggle dataset reflecting the United States labour market circa 2018, whereas the Visual Arts negative-control postings were collected separately by the research team from Asian and global listings during the preparation of this study. Two consequences follow. First, the 2018 vintage means that the corpus

predates the recent expansion of AI- and LLM-related roles. Its most frequent skill terms reflect the data-platform and statistical-computing emphasis of that period—namely, Python, SQL, Machine Learning, R, Hadoop, and Spark (Table 4)—whereas newer competency vocabulary that has since become prominent in data-domain hiring, such as large language models, generative AI, prompt engineering, vector databases, and MLOps, does not appear in the corpus. Consequently, $IF_{c,k}$ reflects the skill emphasis of that period rather than current demand. Second, the temporal, platform, and market gap between the two corpora is a potential confound that we acknowledge directly: because the data-domain and Visual Arts postings differ not only in occupational content but also in collection year, source platform, and geographic market, part of the observed cross-domain separation could, in principle, be attributed to corpus differences rather than to competency differences alone. In addition, the original Indeed dataset was released without a data card, so its search strategy and deduplication procedure cannot be independently verified, platform- or source-selection bias cannot be ruled out, and seniority was not annotated as a separate field. For these reasons the derived $IF_{c,k}$ values should be read as corpus-conditioned structural relevance signals, not as estimates of general or current labour-market demand.

Fifth, $IF_{c,k}$ is currently defined as job-side structural relevance based on the mean path consistency of mapped job-skill records rather than on raw demand frequency. This keeps the score structurally consistent with the ontology, but it also means that the framework deliberately does not reward a knowledge node simply because the associated skill terms appear often. The practical implication is that $IF_{c,k}$ measures how well a node is structurally supported by the ontology path, not how frequently the market demands it, and the two need not coincide.

Finally, the semantic representation diagnostics in Section 5.4 are illustrative rather than a separate classification benchmark. They justify the choice of description-to-definition matching, but they do not independently quantify representation quality against an external gold standard, so the contribution of the augmentation step is supported by ablation and qualitative evidence rather than by a stand-alone accuracy measure.

6.7. Future Work

The limitations above translate into a set of prioritised directions for future research.

First, cross-institutional validation is the most immediate need. Evaluating the framework on transcript data from multiple universities with different credit systems, grading scales, and curriculum structures would clarify which aspects of O4CM performance are specific to the present Thai-university context and which transfer more broadly, directly addressing the single-institution constraint noted above.

Second, occupational and programme expansion towards near-boundary cases is the strongest remaining test of the scoring mechanism. Programmes such as information systems, business analytics, software engineering, digital media, human–computer interaction, computational design, and applied statistics share partial competency overlap with data-domain roles, which makes group separation harder to achieve and therefore more informative than the computing-versus-arts contrast used here. Testing these cases, ideally with individual-level expert-verified ground truth, would reveal whether the PCI mechanism and the $IF_{c,k}$ formulation retain discriminative value once occupational boundaries become less clearly defined and would resolve whether the low within-core separation observed in Section 5.10 reflects real curricular overlap or a limit of the ontology’s discriminative granularity.

Third, the job-posting corpus should be refreshed, and the temporal-market confound should be controlled. Re-running the pipeline on recent postings would capture AI- and LLM-era competency vocabulary that the 2018 corpus omits, while assembling

the contrasting domains from matched-vintage, matched-platform, and matched-market sources would remove the collection-related differences that currently coexist with the competency differences between corpora. This would strengthen the causal interpretation of cross-domain separation and improve the currency of $IF_{c,k}$.

Fourth, LLM augmentation quality control deserves dedicated investigation. Prior work and our own results indicate that LLM-based augmentation can introduce semantic drift and does not always improve ranking accuracy [11]; systematic prompt engineering, post-augmentation validation, and fine-tuned domain-specific encoders are candidate remedies.

Fifth, real hiring-outcome data should be incorporated where available. The current evaluation relies on aggregated expert ratings, which are a useful but imperfect proxy; linking framework-generated rankings to actual placement outcomes, employer feedback, or longitudinal salary data would enable a more ecologically valid assessment of predictive utility.

Sixth, the $IF_{c,k}$ formulation could be extended into a hybrid that combines its present structural-quality definition with raw demand-frequency signals from the job-posting corpora. Such a formulation may better track shifting market demand in dynamic occupational fields, although its behaviour under corpora of differing sizes and skill-frequency distributions would need to be tested explicitly.

Finally, the semantic traceability map should be developed into an interactive decision-support interface, allowing academic advisers, students, and HR practitioners to inspect the course-level evidence behind each ranking, identify curriculum gaps relative to target job types, and compare competency profiles across cohorts. This would strengthen the practical, explainable value of O4CM as an application-oriented competency-mapping system.

Beyond these specific directions, three structural factors may systematically affect O4CM's functioning and should be treated as experimental controls or design variables in future evaluations. First, the diversity of the programme mix directly shapes the range of available student profiles: a cohort drawn from closely related programmes with similar enrolment patterns will produce more similar competency representations, reducing the observable range of NTACS values and making group discrimination harder to demonstrate, regardless of framework design. Second, the GPA distribution—and the grading-quality standards that underlie it—differs across institutions; universities that grade more stringently or more generously will shift the grade-sensitive scoring component in ways that are independent of actual competency differences, and cross-institutional comparisons should account for institutional grading norms when interpreting score-level differences. Third, the design of the job-posting source—including the platform, geographic market, collection period, seniority mix, and deduplication strategy—introduces variation in $IF_{c,k}$ that propagates into every ranking; assembling job-posting corpora under controlled collection conditions is therefore a prerequisite for attributing framework performance to competency structure rather than to data artefacts. These three factors serve as a structural checklist for future replication and comparative studies of O4CM and related competency-mapping frameworks.

7. Conclusions

This paper introduced O4CM, a semi-automated ontology-grounded framework for multi-level competency mapping. Three findings emerge from this evaluation: (1) LLM ensemble induction with HITL verification can produce a scalable and auditable ontology backbone within the evaluated dataset—21 of 22 O*NET knowledge domains reached $\geq 4/5$ consensus, yet HITL correction remained necessary, even for unanimous outputs, demonstrating that model agreement is not a sufficient proxy for domain correctness;

(2) PCI-based multi-level structural scoring achieved bidirectional group separation between computing and arts curricula in the evaluated negative-control setting, outperforming leaf-level cosine similarity in cross-domain discrimination; and (3) the NTACS scoring framework aligns with aggregated expert judgement ($ICC(2,k) = 0.78$) while offering grade-free complementary measures, providing a possible path for institutions where academic grades are absent or unsuitable.

These findings help to address a gap that existing embedding-based methods [11] and manually curated ontologies [4] leave unresolved: individually traceable, job-type-conditioned candidate rankings derived from structured academic transcripts. The semantic traceability map can support explainable evidence inspection for academic advising and exploratory early-career recruitment screening, areas where transparency and accountability are increasingly mandated [6]. Practically, O4CM is designed to be extensible across domains: replacing the O*NET seed taxonomy with a healthcare, engineering, or business knowledge taxonomy may allow for adaptation without retraining the SBERT encoder itself, but this still requires domain-specific corpus collection, LLM ensemble re-execution, HITL re-validation, and empirical revalidation of discriminative performance before deployment.

The study is limited to a single institution and three data-domain roles; the expert reference relies on aggregated ratings rather than verified hiring outcomes, and LLM augmentation can still introduce semantic drift [11] despite HITL safeguards. Cross-institutional validation across diverse curricula and occupational families remains the most immediate priority for future research. Beyond academic validation, O4CM has potential implications for educational and recruitment platforms: universities and accreditation bodies could use the NTACS scores and semantic traceability maps as supplementary evidence for curriculum-gap analysis against the job-posting evidence captured in the target corpus, potentially supporting programme design and graduate employability planning discussions; recruitment platforms could similarly consider the job-type-conditioned ranking as an auditable, explainable screening layer that complements—rather than replaces—existing keyword-based filters, providing course-level evidence to support rather than determine candidate-screening discussions in validated use settings.

Author Contributions: Conceptualization, A.I.-n.; methodology, A.I.-n.; software, A.I.-n.; validation, A.T.; formal analysis, J.B.; investigation, A.I.-n.; resources, A.I.-n.; data curation, A.I.-n.; writing—original draft preparation, A.I.-n.; writing—review and editing, J.B., S.S., and A.T.; visualization, A.I.-n. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by a scholarship from Rajamangala University of Technology Lanna, Thailand, which also covered research-related expenses. The grant number is not applicable.

Institutional Review Board Statement: Ethical review and approval were not required for this study because the research used de-identified student transcript records and aggregated job-posting data. The student records had been transformed before analysis so that individual learners could not be identified, and no personally identifiable information was included in the computational pipeline.

Informed Consent Statement: Informed consent was waived due to the retrospective nature of the study and the use of fully anonymised archival educational records, which posed minimal risk to the participants.

Data Availability Statement: The data presented in this study are not publicly available due to privacy and ethical restrictions regarding student educational records. The data are available upon request from the corresponding author.

Acknowledgments: Aomsap Inkongngarm gratefully acknowledges the support of Rajamangala University of Technology Lanna, Thailand. Use of AI tools: During framework development, AI language models (GPT-5.4, GPT-5.3, Gemini 3 Pro, Claude Opus 4.6, and Claude Sonnet 4.6) were used as components of the LLM ensemble for ontology induction, as described in Section 4.4.3.

AI tools were also used to provide limited editorial assistance during manuscript preparation, including grammar checking and wording suggestions. All ontology mappings produced by the AI components were reviewed and verified by human experts before being accepted into the framework. All scientific content, experimental design, analysis, interpretation, and final manuscript preparation were performed and approved by the authors.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

AI	Artificial Intelligence
BERT	Bidirectional Encoder Representations from Transformers
BIS	Business Information System
CE	Computer Engineering
CS	Computer Science
CV	Curriculum Vitae
DA	Data Analyst
DE	Data Engineer
DS	Data Scientist
ESCO	European Skills, Competences, Qualifications and Occupations
GCS	Grade-Weighted Competency Score (GCS_c)
GPA	Grade Point Average
HITL	Human-in-the-Loop
IE	Information Extraction
IF	Job Importance Factor ($IF_{c,k}$)
IT	Information Technology
KG	Knowledge Graph
LIME	Local Interpretable Model-agnostic Explanations
LLM	Large Language Model
NER	Named Entity Recognition
NTACS	Normalised Total Accumulated Competency Score ($NTACS_c$)
O4CM	Ontology Framework for Multi-level Competency Mapping
O*NET	Occupational Information Network
OWL	Web Ontology Language
PCI	Path Consistency Index
RCV	IF-Weighted Relevant Credit Volume (RCV_c)
RDF	Resource Description Framework
RelVol	Relevant Volume Measure
SBERT	Sentence-BERT
SHAP	SHapley Additive exPlanations
SR	Separation Ratio
TACS	Total Accumulated Competency Score ($TACS_c$)
TF-IDF	Term Frequency–Inverse Document Frequency
VA	Visual Arts
XAI	Explainable Artificial Intelligence

Appendix A. Worked Example: Student A Traced Through the O4CM Framework

Appendix A.1. Purpose and Scope

This appendix traces a single student—hereafter referred to as Student A—through all five phases of the O4CM framework, from transcript preprocessing (Phase 1) to the final ranked score (Phase 5), using actual course records and figures reported in the manuscript.

The goal is to give readers enough detail to reproduce or independently verify the scoring process.

Student A's profile:

- Degree programme: B.Sc. Computer Science;
- Target job type: Data_Scientist (DS);
- Framework outcome: NTACSDS Rank 1 of 94 students.

Data sources: The worked example uses five representative course records for Student A to illustrate the Phase 4 and Phase 5 calculations. These records are used for explanatory purposes to demonstrate the scoring pipeline, including ontology-path locking, PCI_i computation, evidence relevance (E_i), relevance weighting (w_i), and NTACSDS ranking. The level-wise similarity scores (S_1 , S_2 , and S_3) underlying each PCI_i are produced internally by the framework. A complete worked example tracing Student A through all five phases of the O4CM framework is provided in Appendix A, while a detailed step-by-step decomposition for one representative course (Statistics for Science) is presented in Appendix A.5.

Scope: The calculations use the five top-contributing courses. In practice, NTACSc is computed over the full transcript; these five courses illustrate every step and account for the largest share of Student A's score.

Appendix A.2. Worked Example for Phase 1: Atomic Unit Extraction and Preprocessing

Appendix A.2.1. Role of Phase 1 in the Worked Example

Phase 1 converts raw transcript records into clean, standardised atom-level tuples. Three operations are applied: duplicate records are removed, non-informative special characters are stripped, and credit values are converted to numeric form. Each course becomes one atom carrying four fields: Stu_ID, Course, Credits, Grade.

Appendix A.2.2. Student A's Transcript Atoms

Table A1 lists the five courses traced in this example, together with their concise descriptions. All carry three credits and form part of the B.Sc. Computer Science curriculum.

Table A1. Atomic transcript records for Student A selected for the worked example illustrating the Phase 4 and Phase 5 computations. Cr_i denotes the course credits; G_i denotes the numeric grade.

<i>i</i>	Course	Cr_i	Brief Description
1	Statistics for Science	3	Probability, random variables, distributions, sampling, hypothesis testing, ANOVA, chi-squared test, and regression analysis.
2	Fundamental Computer Science	3	Computer evolution, Boolean logic, hardware/software architecture, operating systems, data storage, I/O devices, networks, and introductory flowcharting and programming.
3	Computer Programming	3	Programming fundamentals: data types, variables, expressions, control structures, arrays, procedures, file I/O, and problem-solving practice.
4	Web Programming	3	Web application development covering front-end and back-end programming, client-server architecture, and data-interface design.
5	Cloud Computing	3	Cloud service models, deployment architectures, distributed computing, and cloud-based storage and infrastructure.

After Phase 1, each record is a clean tuple ready for ontology mapping. The box below illustrates the transformation for course 1.

Phase 1 Output for Student A (course $i = 1$)**Before:**

(A001, Statistics for Science, 3 credits, A)

After:(A001, Statistics for Science, $Cr_i=3$, G_i)

Credit value is numeric; no duplicates remain; formatting artefacts removed.

*Appendix A.3. Worked Example for Phase 2: Semi-Automated Bottom-Up Ontology Construction**Appendix A.3.1. Role of Phase 2 in the Worked Example*

Phase 2 constructs the three-level ontology backbone linking student course evidence to job requirements:

- L_1 (Job Categories, 2 nodes): Computer_Job, Art_Job
- L_2 (Skill Groups, 7 nodes): Cognitive Skills, Technical Skills, Creative Skills, Communication Skills, Social and Interpersonal Skills, Psychomotor Skills, Affective Skills
- L_3 (Knowledge Domains, 22 nodes): O*NET knowledge taxonomy

A five-model LLM ensemble proposes Knowledge-to-Skill ($L_3 \rightarrow L_2$) and Skill-to-Job ($L_2 \rightarrow L_1$) mappings. A human expert (HITL) is the final acceptance gate before any mapping is materialised as an `rdfs:subClassOf` axiom. A total of 21 of 22 knowledge domains reached at least 4/5 LLM consensus.

Appendix A.3.2. Ontology Paths Relevant to Student A

The five courses map to two L_3 knowledge domains. Table A2 shows the locked three-level path for each.

Table A2. Expert-validated ontology mappings for the two L_3 knowledge domains relevant to Student A's courses. All `rdfs:subClassOf` links were confirmed during the Phase 2 Human-in-the-Loop verification process.

L_3 Knowledge Domain	L_2 Skill Group	L_1 Job Category
Mathematics	Cognitive Skills	Computer_Job
Computers and Electronics	Technical Skills	Computer_Job

Both paths terminate at Computer_Job, consistent with the HITL-validated ontology (Figure 3 in the main manuscript). Statistics for Science, Computer Programming, and Web Programming lock to the Mathematics path; Fundamental Computer Science and Cloud Computing lock to the Computers and Electronics path.

*Appendix A.4. Worked Example for Phase 3: Semantic Augmentation and Node Representation**Appendix A.4.1. Role of Phase 3 in the Worked Example*

Phase 3 decouples human-readable labels from computational representations by augmenting three types of textual unit and encoding, all with SBERT (all-MiniLM-L6-v2):

1. Ontology nodes (L_1 – L_3): O*NET definitions for L_3 nodes; HITL-validated descriptions for L_2 nodes.
2. Job-side atoms: Each skill atom is expanded in the context of its job posting to clarify competency expectations.

3. Course atoms: Course titles and descriptions are rewritten as competency-oriented statements.

Encoding produces 384-dimensional fixed vectors:

$$\vec{V}_n = f_{\text{SBERT}}(D_n), n \in L_1 \cup L_2 \cup L_3; \quad \vec{V}_u = f_{\text{SBERT}}(T_u), u \in \mathcal{U}.$$

Appendix A.4.2. Augmented Descriptions for Student A's Courses

Table A3 shows the augmented description (T_u) used for each course. Augmentation shifts similarity computation from surface-level keyword overlap toward functional competency meaning.

Table A3. Semantic augmentation applied to Student A's five courses (Phase 3). Augmented descriptions (T_u) are input to SBERT (all-MiniLM-L6-v2) for vector encoding.

Course	Augmented Description (T_u)
Statistics for Science	This course develops competency in statistical reasoning and quantitative data analysis, covering probability, inference, hypothesis testing, regression, and ANOVA. These skills directly support model validation, uncertainty quantification and evidence-based decision-making in data science roles.
Fundamental Computer Science	This course builds foundational knowledge of computer systems, including Boolean logic, hardware and software architecture, operating systems, data storage, and networking. The practical exposure to algorithm design and programming installation provides a structural grounding in how computational systems are organised and operated.
Computer Programming	This course develops competency in structured and procedural programming, covering data types, control flow, arrays, procedures, and file I/O. The algorithmic problem-solving skills gained in this course support the implementation of computational pipelines and automated data workflows.
Web Programming	This course builds competency in Web application development, covering front-end and back-end programming, client-server logic, and data interface design. These programmatic and structural skills are applicable to the deployment of data-accessible services in computing roles.
Cloud Computing	This course develops knowledge of cloud service models, deployment architectures, and distributed computing infrastructure. The technical skills involved in cloud storage and resource management are directly relevant to data engineering and systems-integration roles.

The O*NET definition for Mathematics (L_3) serves as D_n for that node, and the HITL-validated description of Cognitive Skills (L_2) serves as its parent node. After Phase 3, every node and every course atom has a 384-dimensional embedding ready for cosine-similarity computation in Phase 4.

Appendix A.5. Phase 4: Multi-Level Structural Scoring (PCI)

Appendix A.5.1. Role of Phase 4 in the Worked Example

Phase 4 assigns each course a locked ontology path and a Path Consistency Index (PCI_i) measuring alignment across all three ontology levels. Three steps are applied to each course:

Step 1—Entry-point selection at L_3 : The course embedding ($\vec{V}_i$) is compared against all 22 L_3 node embeddings. The best-matching node (k^*) and its leaf-level similarity (S_3) are recorded:

$$k^* = \arg \max_{k \in L_3} \cos(\vec{V}_i, \vec{V}_k), \quad S_3 = \cos(\vec{V}_i, \vec{V}_{k^*}). \quad (\text{A1})$$

Step 2—Branch locking: Parent nodes are retrieved from the validated ontology.

$$g^* = \text{parent}_{L_2}(k^*), \quad j^* = \text{parent}_{L_1}(g^*). \quad (\text{A2})$$

Step 3—Multi-level re-scoring: The course embedding is re-scored against both parent nodes.

$$S_2 = \cos(\vec{V}_i, \vec{V}_{g^*}), \quad S_1 = \cos(\vec{V}_i, \vec{V}_{j^*}). \quad (\text{A3})$$

Path Consistency Index:

$$\text{PCI}_i = \frac{S_1 + S_2 + S_3}{3}. \quad (\text{A4})$$

A high PCI_i indicates alignment at both the leaf level (S_3) and the higher ontology levels (S_2 and S_1). A course with a high S_3 but low S_2 and S_1 is retained but down-weighted.

Appendix A.5.2. WorkingComputation: Statistics for Science ($i = 1$)

Statistics for Science achieves $k^* = \text{Mathematics}$. The locked path is

$$\begin{aligned} \text{Statistics for Science} &\xrightarrow{S_3} \text{Mathematics } (L_3) \\ &\xrightarrow{S_2} \text{Cognitive Skills } (L_2) \\ &\xrightarrow{S_1} \text{Computer_Job } (L_1). \end{aligned}$$

Table A4. Level-wise cosine similarity scores and the resulting Path Consistency Index (PCI) for Statistics for Science ($i = 1$, Student A). The aggregate $\text{PCI}_1 = 0.481$ is computed from the three level-wise cosine similarity scores shown in this table.

Course	S_3	S_2	S_1	$\text{PCI}_i = (S_1 + S_2 + S_3)/3$
Statistics for Science	0.582	0.456	0.405	$(0.582 + 0.456 + 0.405)/3 = 0.481$

$S_3 = 0.582$ reflects the close match between the augmented Statistics description and the O*NET Mathematics definition. $S_2 = 0.456$ confirms contextual support from Cognitive Skills, and $S_1 = 0.405$ confirms partial alignment with Computer_Job. The consistency across all three levels means $\text{PCI}_1 = 0.481$ reflects genuine hierarchical support rather than an isolated leaf-level coincidence.

Appendix A.5.3. PCI Values for All Five Courses

The same procedure is applied to every course in the transcript. Table A5 summarises the locked ontology paths and the corresponding PCI_i values for the five representative courses selected for this worked example.

Table A5. Locked ontology paths and corresponding Path Consistency Index (PCI_i) values for the five representative courses used in the worked example. The PCI_i values are those obtained from the Phase 4 ontology-path locking procedure.

i	Course	L_3 Node k^*	Cr_i	PCI_i
1	Statistics for Science	Mathematics	3	0.481
2	Fundamental Computer Science	Computers and Electronics	3	0.463
3	Computer Programming	Mathematics	3	0.457
4	Web Programming	Mathematics	3	0.445
5	Cloud Computing	Computers and Electronics	3	0.432

Courses 1, 3, and 4 lock to Mathematics via Cognitive Skills. Courses 2 and 5 lock to Computers and Electronics via Technical Skills. PCI_i ranges from 0.432 to 0.481, reflecting consistent hierarchical support across all five records.

Appendix A.6. Phase 5: Job-Conditioned Scoring and Candidate Ranking

Appendix A.6.1. Role of Phase 5 in the Worked Example

Phase 5 combines the student-side structural scores (PCI_i) from Phase 4 with a job-side structural relevance factor ($IF_{c,k}$) derived from job-posting evidence, producing TACS, NTACS, and a course-level traceability signal (E_i).

Appendix A.6.2. Step 1: Importance Factor ($IF_{c,k}$)

Requirement mass: For each job type (c) and L_3 node k , PCI values of job-skill atoms mapping to node k are aggregated:

$$m_{c,k} = \sum_{r \in \mathcal{R}_{c,k}} PCI_r. \quad (A5)$$

Requirement share:

$$REQ_{c,k} = \frac{m_{c,k}}{\sum_{k' \in K_c} m_{c,k'}}, \quad (A6)$$

where K_c is the set of L_3 nodes under c 's L_1 category.

Job-type specificity:

$$\overline{REQ}_k = \frac{1}{|C|} \sum_{c'} REQ_{c',k}, \quad Spec_{c,k} = \log \left(1 + \frac{\max(0, REQ_{c,k} - \overline{REQ}_k)}{\varepsilon + \overline{REQ}_k} \right). \quad (A7)$$

Normalised Importance Factor:

$$IF_{c,k} = \frac{REQ_{c,k} \times Spec_{c,k}}{\max_{k'} (REQ_{c,k'} \times Spec_{c,k'})}. \quad (A8)$$

Appendix A.6.3. Deriving the Job-Type Importance Factor ($IF_{DS,k}$)

For the worked example, the job-type importance factor is obtained by rearranging the evidence-relevance equation defined in Equation (15). Solving for $IF_{c,k(i)}$ gives

$$IF_{c,k(i)} = \frac{E_i}{PCI_i}. \quad (A9)$$

Using one representative course associated with each knowledge node:

- Mathematics (from Statistics for Science, $i = 1$):
 $IF_{DS,Math} = 0.342/0.481 \approx 0.711$;
- Computers and Electronics (from Fundamental Computer Science, $i = 2$):
 $IF_{DS,C\&E} = 0.319/0.463 \approx 0.689$.

Mathematics ($IF = 0.711$) is the most job-specifically demanded node for Data Scientist, consistent with the statistical and algorithmic nature of the role. Computers and Electronics ($IF = 0.689$) reflects the programming and systems requirements.

Cross-check via Computer Programming and Web Programming (both map to Mathematics): The calculations demonstrate the expected consistency across both courses, yielding a comparable ratio of approximately 0.71 in both cases:

$$\frac{E_3}{\text{PCI}_3} = \frac{0.325}{0.457} \approx 0.711, \quad \frac{E_4}{\text{PCI}_4} = \frac{0.316}{0.445} \approx 0.710.$$

Both calculations confirm $\text{IF}_{\text{DS,Math}} \approx 0.711$

Appendix A.6.4. Step 2: Evidence-Relevance Signal E_i

$$E_i = \text{PCI}_i \times \text{IF}_{c,k(i)}. \quad (\text{A10})$$

Table A6 reproduces and verifies all five E_i values.

Table A6. Evidence-relevance signals (E_i) for the five representative courses used in the worked example. The reference E_i values are those used by the framework, while the derived values are independently recomputed from $\text{PCI}_i \times \text{IF}_{c,k(i)}$ to verify the consistency of the calculation.

i	Course	PCI_i	$\text{IF}_{\text{DS},k(i)}$	E_i (Framework)	E_i (Derived)
1	Statistics for Science	0.481	0.711	0.342	$0.481 \times 0.711 = 0.342$
2	Fundamental Computer Science	0.463	0.689	0.319	$0.463 \times 0.689 = 0.319$
3	Computer Programming	0.457	0.711	0.325	$0.457 \times 0.711 = 0.325$
4	Web Programming	0.445	0.711	0.316	$0.445 \times 0.711 = 0.316$
5	Cloud Computing	0.432	0.689	0.298	$0.432 \times 0.689 = 0.298$

Appendix A.6.5. Step 3: Relevance Weight (w_i) and TACS

$$w_i = Cr_i \times \text{PCI}_i \times \text{IF}_{c,k(i)}. \quad (\text{A11})$$

Table A7. Relevance weights (w_i) for Student A's five top-contributing courses.

Course	Cr_i	PCI_i	$\text{IF}_{\text{DS},k(i)}$	w_i
Statistics for Science	3	0.481	0.711	$3 \times 0.481 \times 0.711 = 1.026$
Fundamental Computer Science	3	0.463	0.689	$3 \times 0.463 \times 0.689 = 0.957$
Computer Programming	3	0.457	0.711	$3 \times 0.457 \times 0.711 = 0.975$
Web Programming	3	0.445	0.711	$3 \times 0.445 \times 0.711 = 0.949$
Cloud Computing	3	0.432	0.689	$3 \times 0.432 \times 0.689 = 0.893$
Sum $\sum w_i$ (top 5):				4.800

The raw TACS accumulates grade-weighted evidence over the full transcript:

$$\text{TACS}_c = \sum_{i=1}^n G_i \times w_i. \quad (\text{A12})$$

The high w_i values, combined with Student A's high grades (consistent with Rank 1 of 94), already position Student A at the top before normalisation.

Appendix A.6.6. Step 4: Normalised TACS (NTACS_c) and Ranking

$$\text{NTACS}_c = \frac{\sum_{i=1}^n G_i \times w_i}{\sum_{i=1}^n w_i}, \quad w_i = Cr_i \times \text{PCI}_i \times \text{IF}_{c,k(i)}. \quad (\text{A13})$$

NTACS_c is a credit-weighted grade average in which each course is weighted by structural alignment (PCI) and job-side relevance (IF), not credits alone.

Numerical illustration: Using only the five courses above with $G_i = 4$ (top grade, 0–4 scale),

$$\sum_{i=1}^5 G_i w_i = 4 \times 4.800 = 19.200,$$

$$\sum_{i=1}^5 w_i = 4.800,$$

$$\text{NTACS}_{\text{DS}} (\text{5-course}) = 19.200 / 4.800 = \mathbf{4.00}.$$

Over the full transcript, Student A achieves the highest NTACS_{DS} in the cohort: Rank 1 of 94.

Appendix A.6.7. Relevant Volume Measure

$$\text{RelVol} = \sum_{i=1}^n Cr_i \times \text{PCI}_i. \quad (\text{A14})$$

For the five illustrative courses,

$$\text{RelVol} (\text{top 5}) = 3 \times (0.481 + 0.463 + 0.457 + 0.445 + 0.432) = 3 \times 2.278 = 6.834.$$

This grade-free indicator measures the total credit volume supported by the locked ontology paths.

Appendix A.7. Semantic Traceability Map

Table A8 is the semantic traceability map for Student A: each row traces a course through its locked ontology path to its contribution to the Data Scientist ranking.

Table A8. Semantic traceability for the five representative courses used in the worked example (Data Scientist job type). The PCI_i , $\text{IF}_{c,k(i)}$, and E_i values are those obtained in the worked example, while the credit-weighted contributions ($Cr_i \times E_i$) are computed directly from the corresponding course credits and evidence-relevance scores.

Course	L_3 Node	L_2 Node	Cr_i	PCI_i	E_i	$Cr_i \times E_i$
Statistics for Science	Mathematics	Cognitive Skills	3	0.481	0.342	1.026
Fundamental Computer Science	Comp. & Elec.	Technical Skills	3	0.463	0.319	0.957
Computer Programming	Mathematics	Cognitive Skills	3	0.457	0.325	0.975
Web Programming	Mathematics	Cognitive Skills	3	0.445	0.316	0.948
Cloud Computing	Comp. & Elec.	Technical Skills	3	0.432	0.298	0.894
Total $\sum Cr_i \times E_i$ (worked example):						4.800

A reader or advisor can follow any row (course $\rightarrow L_3 \rightarrow L_2 \rightarrow L_1$) to verify why each course receives its weight in the ranking.

Appendix A.8. Summary

Table A9. Phase-by-phase outputs for Student A (B.Sc. Computer Science, NTACS_{DS} Rank 1 of 94).

Phase	Operation	Output for Student A
Phase 1	Atomic extraction	Five course atoms: Statistics for Science, Fundamental Computer Science, Computer Programming, Web Programming, and Cloud Computing ($Cr_i = 3$ for all).

Table A9. Cont.

Phase	Operation	Output for Student A
Phase 2	Ontology induction & HITL	Two L_3 paths materialised: Mathematics $\rightarrow$ Cognitive Skills $\rightarrow$ Computer_Job; Comp. & Elec. $\rightarrow$ Technical Skills $\rightarrow$ Computer_Job.
Phase 3	SBERT augmentation & encoding	All five course descriptions expanded into competency statements and encoded as 384-dimensional vectors (all-MiniLM-L6-v2).
Phase 4	Multi-level structural scoring	Locked paths confirmed; PCI_i : 0.481, 0.463, 0.457, 0.445, 0.432 (all ≥ 0.43 , consistent hierarchical support).
Phase 5	Job-conditioned scoring & ranking	$IF_{DS,Math} = 0.711$; $IF_{DS,C\&E} = 0.689$. Weights w_i : 1.026, 0.957, 0.975, 0.949, 0.893. E_i : 0.342, 0.319, 0.325, 0.316, 0.298. Full-transcript NTACSDS: Rank 1 of 94.

The consistency of high PCI_i values (all ≥ 0.43) and high IF weights for the two most-demanded knowledge nodes (Mathematics and Computers & Electronics) explains Student A's top-ranked position. Every weight in Equation (A13) traces back to a specific course, a specific ontology path, and a specific job-posting evidence source.

References

- Decorte, J.J.; Verlinden, S.; Van Haute, J.; Deleu, J.; Develder, C.; Demeester, T. Extreme multi-label skill extraction training using large language models. *arXiv* **2023**, arXiv:2307.10778.
- Massenkoff, M.; McCrory, P. *Labor Market Impacts of AI: A New Measure and Early Evidence*; Technical Report; Anthropic: San Francisco, CA, USA, 2026.
- Reimers, N.; Gurevych, I. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2019; pp. 3982–3992. [\[CrossRef\]](#)
- Trajanoska, M.; Stojanov, R.; Trajanov, D. Enhancing knowledge graph construction using large language models. *arXiv* **2023**, arXiv:2305.04676.
- Ji, Z.; Lee, N.; Frieske, R.; Yu, T.; Su, D.; Xu, Y.; Ishii, E.; Bang, Y.J.; Madotto, A.; Fung, P. Survey of hallucination in natural language generation. *ACM Comput. Surv.* **2023**, *55*, 1–38. [\[CrossRef\]](#)
- Doshi-Velez, F.; Kim, B. Towards a rigorous science of interpretable machine learning. *arXiv* **2017**, arXiv:1702.08608.
- Adadi, A.; Berrada, M. Peeking inside the black-box: A survey on explainable artificial intelligence (XAI). *IEEE Access* **2018**, *6*, 52138–52160. [\[CrossRef\]](#)
- Zhang, M.; Jensen, K.; Sonniks, S.; Plank, B. SkillSpan: Hard and soft skill extraction from English job postings. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2022; pp. 4962–4984.
- Clavié, B.; Ciceu, A.; Naylor, F.; Soulié, G.; Brightwell, T. Large language models in the workplace: A case study on prompt engineering for job type classification. In *Proceedings of the International Conference on Applications of Natural Language to Information Systems*; Springer: Berlin/Heidelberg, Germany, 2023; pp. 3–17.
- Peterson, N.G.; Mumford, M.D.; Borman, W.C.; Jeanneret, P.R.; Fleishman, E.A.; Levin, K.Y.; Campion, M.A.; Mayfield, M.S.; Morgeson, F.P.; Pearlman, K.; et al. Understanding Work Using the Occupational Information Network (O*NET): Implications for Practice and Research. *Pers. Psychol.* **2001**, *54*, 451–492. [\[CrossRef\]](#)
- Inkong-ngarm, A.; Bootkrajang, J.; Somhom, S.; Trongratsameethong, A. Student-to-job Ranking Framework Based on Knowledge Space Embedding. In *Proceedings of the 2025 International Conference on Educational Technology Management (ICETM)*; IEEE: New York, NY, USA, 2025; pp. 141–145.
- Malinowski, J.; Keim, T.; Wendt, O.; Weitzel, T. Matching people and jobs: A bilateral recommendation approach. In *Proceedings of the 39th Annual Hawaii International Conference on System Sciences (HICSS'06)*; IEEE: New York, NY, USA, 2006; Volume 6, p. 137c.
- Faliagka, E.; Ramantas, K.; Tsakalidis, A.; Tzimas, G. Application of machine learning algorithms to an online recruitment system. In *Proceedings of the International Conference on Internet and Web Applications and Services*, Stuttgart, Germany, 27 May–1 June 2012; pp. 215–220.
- Çelik Ertuğrul, D.; Bitirim, S. Job recommender systems: A systematic literature review, applications, open issues, and challenges. *J. Big Data* **2025**, *12*, 140. [\[CrossRef\]](#)

15. Nasar, Z.; Jaffry, S.W.H.; Malik, M.K. Named Entity Recognition and Relation Extraction: State-of-the-Art. *ACM Comput. Surv.* **2021**, *54*, 1–39. [[CrossRef](#)]
16. Ternikov, A. Soft and hard skills identification: Insights from IT job advertisements in the CIS region. *PeerJ Comput. Sci.* **2022**, *8*, e946. [[CrossRef](#)] [[PubMed](#)]
17. Lukauskas, M.; Šarkauskaitė, V.; Pilinkienė, V.; Stundžienė, A.; Grybauskas, A.; Bruneckienė, J. Enhancing skills demand understanding through job ad segmentation using NLP and clustering techniques. *Appl. Sci.* **2023**, *13*, 6119. [[CrossRef](#)]
18. U.S. Department of Labor. O*NET Resource Center. 2010. Available online: <https://www.onetcenter.org/> (accessed on 1 November 2025).
19. European Commission. *ESCO: European Skills, Competences, Qualifications and Occupations*; European Commission: Brussels, Belgium, 2017.
20. Akundi, A.; Ravipati, P.R.T.; Luna Fong, S.A.; Otieno, W. Industry-Driven Model-Based Systems Engineering (MBSE) Workforce Competencies—An AI-Based Competency Extraction Framework. *Systems* **2025**, *13*, 781.
21. Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. Efficient estimation of word representations in vector space. *arXiv* **2013**, arXiv:1301.3781.
22. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2019; pp. 4171–4186.
23. Decorte, J.J.; Van Haute, J.; Demeester, T.; Devellder, C. Jobbert: Understanding job titles through skills. *arXiv* **2021**, arXiv:2109.09605.
24. Ren, X.; Tang, J.; Yin, D.; Chawla, N.; Huang, C. A survey of large language models for graphs. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*; ACM: New York, NY, USA, 2024; pp. 6616–6626.
25. Jemal, I.; Armand, N.S.W.; Chikhaoui, B. A new approach for competency frameworks mapping using large language models. *Expert Syst. Appl.* **2025**, *263*, 125648.
26. Artificial Analysis. AI Model Leaderboard: Independent Analysis of AI Models and API Providers. 2026. Available online: <https://artificialanalysis.ai/leaderboards/models> (accessed on 15 December 2025).
27. Ribeiro, M.T.; Singh, S.; Guestrin, C. “Why should i trust you?” Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*; ACM: New York, NY, USA, 2016; pp. 1135–1144.
28. Lundberg, S.M.; Lee, S.I. A unified approach to interpreting model predictions. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 4768–4777.
29. Chen, J.; Zhang, C.; Niu, Z. A Two-Step Resume Information Extraction Algorithm. *Math. Probl. Eng.* **2018**, *2018*, 5761287. [[CrossRef](#)]
30. Javadian Sabet, A.; Bana, S.H.; Yu, R.; Frank, M.R. Course-Skill Atlas: A national longitudinal dataset of skills taught in US higher education curricula. *Sci. Data* **2024**, *11*, 1086. [[PubMed](#)]
31. elroyggj. *Indeed Dataset: Data Scientist, Data Analyst, Data Engineer*; Kaggle: San Francisco, CA, USA, 2018.
32. Musen, M.A. The protégé project: A look back and a look forward. *AI Matters* **2015**, *1*, 4–12. [[PubMed](#)]
33. Yun, Y.; Cao, R.; Dai, H.; Zhang, Y.; Shang, X. Self-paced graph memory for learner GPA prediction and it’s application in learner multiple evaluation. *Sci. Rep.* **2023**, *13*, 21407. [[PubMed](#)]
34. Deshmukh, A.; Raut, A. Enhanced Resume Screening for Smart Hiring Using Sentence-Bidirectional Encoder Representations from Transformers (S-BERT). *Int. J. Adv. Comput. Sci. Appl.* **2024**, *15*, 241. [[CrossRef](#)]
35. Mann, H.B.; Whitney, D.R. On a test of whether one of two random variables is stochastically larger than the other. In *The Annals of Mathematical Statistics*; Institute of Mathematical Statistics: Beachwood, OH, USA, 1947; pp. 50–60.
36. CC2020 Task Force. *Computing Curricula 2020*; ACM: New York, NY, USA, 2020.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.