Cloud-Based Mechanized Method for Developing Semantically Rich Ontologies and Planning Analogies for E-comfort Bids

*Ms. Dimpy Jindal, **Prof. Manju Kaushik, ***Dr. Barkha Bahl

ABSTRACT

The ontology framework at the temperament of the semantic web is a powerful method for representing and visualising domain knowledge. Estimating similarity measures between ontologies, determining a threshold, and employing if-then procedures to validate relevance and irrelevance all contribute to the reusability of knowledge. Knowledge visualisation at a reduced level is supplied by simplified semantic representations of the ontology, which is particularly useful for processing and analysing e-health data. Resolving implicit knowledge, which often develops in the attendance of implicit information and polymorphic objects and manifests as non-dominant words and conditionally dependent actions, enables the creation of semantically complex constructs. In this study, we clarify in detail how the automated system constructs and stores ontology structures rich in semantics. Graph Derivation Representation, which is based on dyadic deontic logic, is used to construct ontologies with a high density of meaning. In addition, the usual cosine similarity metric is used to determine the degree of similarity between two ontologies. In response to a document stored in the cloud, basic if-then rules are used to count how many relevant documents there are and retrieve their corresponding metadata. These functional modules are used in e-health applications for document recovery, information removal, and domain dictionary generation, and they will be of great help to authenticated cloud users. According to the diabetes dataset experiments, the suggested framework outdoes the state-of-the-art Graph Derivation Representation methods. The visual representations of the paper's findings provide another perspective for evaluating the usefulness of the proposed methodology.

KEYWORDS

Ontology framework, Semantic web, Knowledge visualisation, E-health data, Graph Derivation Representation, Dyadic deontic logic, Cosine similarity, Cloud-based document retrieval

*Assistant Professor, Delhi Institute of Advanced Studies, Delhi, India
**Professor at Amity Institute of Information Technology, Rajasthan, India
*** Ex-Director, Trinity Institute of Professional Studies, Delhi, India

INTRODUCTION

Cloud computing services are a dramatic break from the traditional methods of providing a company's infrastructure, platform, and software with a broad variety of services. The market is booming because of the various benefits it offers, including as flexibility and a pay-as-you-go model. This thorough explanation and the main benefits are provided by NIST in [17]. Computing in the cloud, or "the cloud," refers to a paradigm of delivering IT services in which users pay for actual use of a shared, elastic pool of adjustable hardware and software components. This kind of pay-as-you-go service may be developed and administered with little involvement from either technical management or the cloud service provider [17]. Important qualities that cloud services should enable include scalability, a pay-as-you-go pricing model, a decentralised architecture, a high degree of security, and the ability to virtualize [1]. Virtualized infrastructure, elastic capacity, persistent connection, and pay-as-you-go pricing are just some of the benefits of cloud computing services. Several more advantages are made possible by this shift in the way businesses are managed.

There is a close connection between the semantic structure of the term "ontology," which means "theory of being," and its literal translation. The ability to utilise this semantic structure as the foundation for a knowledge-sharing framework that allows for the representation, diffusion, and reuse of domain expertise is a significant advantage [9]. Ontologies have found applications in many different fields, such as knowledge management, information retrieval, the Semantic Web, information integration, semantic search, and recommendation systems. In this essay, we draw the conclusion that Ontology as a Service is the most important facet of cloud computing because of the influence it has on Infrastructure as a Service. However, Ontology as a Service and its applications were first introduced in [27]. According to the authors of the aforementioned study, "ontology as a service" (OaaS) is "a service where Cloud service providers deploy the ontology creation application and infrastructure together depending on the customers' needs." We utilize the cloud server to create an ontology for the text documents that verified cloud users have uploaded, and then we use an ontology alignment procedure to predict which papers are connected to each other. To facilitate this process, the cloud service provider is used.

Syntactic and semantic information of the intended input text may be communicated using a variety of knowledge representation languages used in artificial intelligence [14]. The expressiveness of the structure is crucial to the initial stage of ontology creation. Ontologies may be represented in a variety of logical systems, including predicate logic, fuzzy logic, temporal logic, situational logic, description logic, and modal logic [23]. Frequently, these applications will use the usage of Description Logic (DL) to store and convey information. However, because to the presence of non-dominant words in the target datasets, DL for faultless and expressive structure is impractical for certain applications. Instead, the insights from the data can only be presented correctly if the structure is expressive. Problems like instability and incompleteness arise because of the reduced expressivity of the intended data. Having polymorphic items in the dataset is already a challenging problem due to the expressivity being a major key issue [16]. The use of modal logic, which includes relative pronouns and events with varying probabilities based on their outcomes, may help one express themselves more fully by eliciting dormant semantic understanding. Dyadic Deontic reasoning is a kind of modal logic that gives more weight to words that aren't often utilised in written discourse. There is a formal discipline dedicated to the study of required, banned, permissible, conditional mandatory, and conditional permissible clauses. It is able to interpret conditional dependency statements and phrases including SHOULD NOT, MUST NOT, SHALL NOT, COULD NOT, and WILL NOT in lieu of the conventional negation symbols used in other logic languages. It also includes the supplementary notations of description logic.

The second crucial aspect of ontology is the aptitude to recycle the produced ontology, meanwhile it is time-consuming to develop new ontology from scratch each time. This idea of recycling the semantics is known as "ontology reuse." The procedure of ontology reuse [24] allows the semantic content of a current ontology to be included into a newly built ontology, even in a different situation. In order to determine the extent of the duplication, an evaluation of the material's reusability is a vital requirement. Calculational measurements of similarity and intersection may be useful here. There are a variety of distance metrics available in the literature that may be used to compute the grade of resemblance amid two ontology systems.

Need of the hour - semantics

The procedure of automating the transfer, exchange, and reuse of data or information through the Internet is crucial but often challenging. As a result of their lack of semantics, HTML, XML, and the URLs they use are sometimes viewed as "dead ends." [12] means that despite the progress in information technology, the above difficulties have extremely limited use on the web. Syntactic and structural heterogeneity issues can be addressed with a amount of approaches that have been documented in the literature [21]. Nonetheless, getting around the issue of semantic heterogeneity is notoriously difficult. An issue known as semantic heterogeneity arises when different contexts do not agree on the same meaning of a piece of data. Synonymous groups, idea lattices, distinguishing characteristics, and limiting predicaments are all examples of semantic heterogeneity issues [18]. These issues have been, at least partially, resolved in the past. Effective methods, however, are required to find a long-term solution to this issue.

Reusability – degree of similarity

Issues of semantic heterogeneity may be overcome with the help of ontology structure.
The process of Ontology Alignment in the semantic web involves the use of ontology reuse metrics to establish semantic correspondences between conceptually similar items in different Ontologies. Ontology and the ontology alignment procedure that follows are used extensively in knowledge management [5, 8], e-commerce [11], e-learning [19], and information retrieval [12], as well as in semantic search and recommendation systems [11].

Ontology arrangement is becoming more and more important and time-consuming as the ontology system expands in size and complexity. Since this is the case, automated ontology alignment has gained traction in a wide variety of practical applications, comprised of IR/RS, online retail and education, query processing, data/information integration, and transformation, and online retail. Some of the methods for Ontology Alignment that can be found in the books are those that use Strings, Tongues, Restraints, and Semantics. [7,9]. Hwever, there are two major drawbacks to current Ontology Alignment methods in the literature:

1. Abridged semantic articulateness of the built ontology,
2. The most frequently occurring terms in the incoming text documents are used to retrieve the ideas, relationships, axioms, and path linkages from the current frameworks. So, it's important to supply smart methods for efficient Ontology Arrangement, with the goal of ontology recycle.

Objectives

In this research, we offer an automatic outline that features individual functional modules for building ontologies, evaluating their expressivity, and determining how similar they are to one another. The cloud provider can utilise this similarity calculation to serve up relevant files to verified users. For this purpose, we employ a threshold value in the similarity degree and common if-then methods for retrieving related documents. The ontology construction module employs a GDR (Graph Derivation Representation) technique based on dyadic deontic logic to create a semantically rich, evocative ontology. The suggested framework consists of four distinct stages. In the first stage, cloud users are authenticated in the usual way, with a username and password. After that, the verified cloud customers upload their unprocessed but still significant papers to the cloud storage provider. In order to build an ontology with a lot of expressive power, we first take the raw texts and transform them into a dyadic rule's representation. Second, a GDR is created for each concept, connection, and occurrence in the ontology. As a result of the recursive nature of graphical derivations, this is made easier. Later, the various graph node structures are merged using an integration technique to yield an early unified GDR for the provided ontology. After the unstable associations necessary for semantic measurements are removed, a full GDR picture of the provided ontology is produced. In the final stage, we calculate the ontology's semantic expressivity factor and use the cosine similarity metric to determine how similar two ontology structures are to one another. The final step involves retrieving the relevant documents and making them available to authorised cloud users. The threshold estimation module and standard if-then rule building make this possible. Listed below are some of the primary goals of the suggested structure:

- In order to make it easier for verified cloud customers to upload raw text documents to their cloud service provider.
- Using GDR, offer a stable ontological framework for the underlying knowledge. In other words, show how hidden information, underused words, and the likelihood of certain events may be used to build an ontology with a lot of expressive power.
- To quantify the extent to which the semantically dense ontology framework may convey ideas. Determine how similar two ontologies are by a calculation and cosine-similarity-based structures.
- The rule's metric is used to obtain the metadata for as- sociated documents and make it available to authorized cloud users.

Quick analysis on the objectives

The suggested framework has six main goals. The next discussion is an example-based, step-by-step breakdown of the goals.

First, the users should ideally be healthcare professionals, as this paper makes use of a diabetic dataset. The paper's cloud users, who include doctors, nurses, lab technicians, the hospital dean, etc., can save text documents pertaining to patient records in the cloud. Documents uploaded to the cloud can be in any arrangement desired and uploaded by any authorised user. If a cardiologist wants to share information about new tools for diagnosing heart illness or performing heart surgery, they can upload a document describing the topic.

Second, this provided paper may be quite large, and its technical details should be of use to anyone, regardless of whether or not he works in the healthcare industry. Many terms associated with heart disease and cardiac surgery, for instance, might be included in the paper. In order to glean the hidden information from the text, a sematic knowledge picture must be constructed. The research provides a theoretical basis for the creation of ontologies.

Step 3: Several methods exist for generating ontology structure, which can be used in conjunction with Step 2 to further clarify the situation. When conditional probability events occur, however, or when non-dominant arguments such as can, will, cannot, may not, etc. are used, some of the logic representations will be inaccurate. In the suggested framework, an ontology structure with extensive expressiveness is created.

In Step 4, we count the number of classes, relationships, and instances, among other things, so that we may evaluate how well different logic representations convey the underlying data. When compared to more traditional methods, these figures are significantly greater in the suggested framework.

Having determined the documents that are similar (or different), the next step is to compare them. The resemblance between these two submitted documents can be identified in the event that multiple cardiologists are uploading different materials potentially in the same domain. This is very important for the time when a new skill called ontology amalgamation is implemented. A subfield of information picture, ontology merging can be used to combine two medical records that are essentially the same but uploaded by different doctors. However, the topic of combining ontologies is avoided here. Currently, the suggested framework can only be used to determine how similar two papers are to one another.

Here's Step 6: Founded on the user's contribution text, the general users of this framework can pull up a plethora of related papers. A basic if-then rules classifier is employed for retrieval, and similarity computation is crucial for procurement the pertinent IDs. All of the documents' metadata are retrieved and presented to the authorised cloud user. A physician, for example, can upload a single document and then access the metadata of numerous linked papers for the sake of study or record-keeping.

The remaining parts of this paper will be organised as shadows. The second unit offers a brief overview of the pertinent literature. The suggested framework is labelled in depth in Section 3. The suggested framework's performance evaluation is covered in Section 4. The final chapter provides some final thoughts and suggestions for where to take things moving forward.

Literature review

For the purpose of building ontologies and computing measures of similarity, there are a plethora of graphical models available at present [3]. Object Constrained Language (OCL), which is built on top of UML, is one such method. To graphically depict ontologies, OCL is utilised. Instead of describing implicit (hidden) non-taxonomic relationships, UML is well-suited to describe explicit taxonomical information [12]. Semantic Link Network (SLN) is an approach for discovering descriptions of semantic links between pre-existing things. Rather of focusing on semantic correctness, SLN aids prioritise the property of semantic richness [20].

Ontology measurement is a method of evaluating ontologies based on the measures themselves, and current ontology measures rely solely on the information displayed by ontologies to evaluate the degree of resemblance between ontological entities and structures. In the academic literature, cluster-based algorithms are utilised, which, by combining the least path length and the taxonomic depth, define bunches for apiece of the twigs by admiration to the root node. It is often suggested to utilize an ontology-based measure that makes use of taxonomic properties, but does so without relying on tuning parameters to alter the weights assigned to individual features. [13].

To find a related class of ideas, we utilise a semantic matching approach [2] to pull out the relevant concepts' super- and sub-concepts and then apply a similarity function. The suggested technique uses the terms from a graph-based ontology to measure the degree of resemblance between two gene products. Quality measurements may be used to assess and compare a number of different aspects of ontologies, including their expressiveness, cohesion, complexity, richness, and grade of resemblance [14-18]. The complexity of ontology-represented polymorphic objects makes them challenging to manage inside most pre-existing system frameworks. Our study details the steps needed to create an automated framework for building an ontology structure that is both robust and expressive, while also being capable of efficiently managing polymorphism in its representation of ontologies. Thereby providing a solution to this problem. The framework's primary focus is on predicting the degree of resemblance among two ontology systems with the goal of facilitating their reuse.

Analysis of earlier works

The following characteristics must be included in the graphical model that represents the target dataset.

• Semantic information, such as the hidden relationship between ideas and other kinds of connections that can't be captured by traditional taxonomies, has to be conveyed precisely and efficiently. It is significant that the model be able to be assessed using standard ontology metrics.

• The ontology creation framework needs to be flexible enough to incorporate under-represented concepts and proceedings with respect to provisional probability.

• To maintain the integrity of the ontology, the model must take into account the challenge of representing polymorphic items.

• It must be possible to automatically compute the degree of similarity value and incorporate this into the automated framework; this is a crucial requirement for determining the articulateness of the ontology construction.

However, the majority of currently available graphical models that are presented in the literature review don't meet the aforementioned requirements. As a result, a novel approach to generating a GDR that captures both the hidden implicit information and the explicit knowledge must be developed. Algorithm development to address polymorphism in explicit and implicit knowledge representation is also crucial [10].

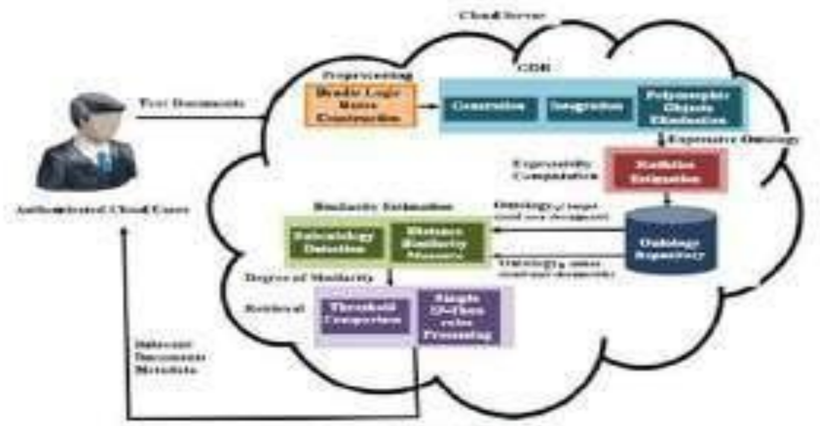

Fig. 1. Proposed automated framework

The articulation of an ontology may be done in any number of information picture languages, such as logic, frames, semantic nets, etc. The bulk of previous studies on the subject of ontology representation have concentrated on the use of logic as the language for describing expertise. In order to attribute the existence of implicit knowledge, we look for terms that are underrepresented in the target data set. It is also conceivable to run across sentences that include both non-dominant words and conditionally likely occurrences. The purpose of this study is to increase expressivity by separating conditionally probable occurrences from dominant and non-dominant words and processing them separately. In addition, for the sake of ontology reuse, certain metrics of resemblance calculation will be necessary in the future.

Feature	Property
Knowledge Representation Languages	Utilizes dyadic logic rules and construction Goes beyond the focus on logic from previous studies
Underrepresented Terms	Identifies underrepresented terms in the tar- get dataset- Attributes the presence of implicit knowledge
Conditionally Probable Events	Separates conditionally probable events from dominant and non-dominant words- Processes conditionally probable events separately
Stability Measurement	Generates stable and semantically rich ontologies using the GDR-DYDL approach
Expressivity Enhancement	Achieved by separating and processing condi- tionally probable events, dominant words, and non-dominant words
Similarity Calculation Metrics	Required for ontology reuse in the future Aligns with the importance of ontology reuse highlight- ed in the text
Performance Comparison	Outperforms the baseline method on the medi- cal diabetes dataset

This table organizes the features and properties of the proposed automated framework, making it easier to understand its key aspects and how they contribute to addressing the limitations of previous studies while enabling the generation of stable, semantically rich, and reusable ontologies.

Proposed system framework

Genuine cloud users can expect to receive extremely relevant materials in reply to their supplied raw documents. In the proposed automated system, GDR serves as a graphical representation of semantic descriptions of textual content. Ontologies are measured and compared using their underlying GDR for consistent semantic assessment, which is why GDRs are generated for them. The full structural semantics of the target ontology can be derived and understood with its assistance. After an ontology has been successfully generated with the use of the GDR method and dyadic rules cohort, the structure of the ontology should be examined for its level of expressivity. This expressivity metric is useful for determining how well the language represents its implicit knowledge. By first representing knowledge in dyadic deontic logic, then translating that into the equivalent GDR, we can get greater expressivity.

The suggested automated approach also makes it easier to calculate how similar two ontologies are to one another. The cosine similarity metric is used for this purpose. After doing a series of trials on the underlying dataset, the threshold value is estimated, and the aforementioned methods for retrieving the associated documents based on similarity value are put into practise. Accordingly, the proposed automated approach provides a practical means of developing semantically rich ontology structures, evaluating expressivity, employing a rule-based metric to approximation the degree of resemblance between two documents and retrieve them if they are found to be related. This automated framework can be used for a variety of purposes, such as recommendation systems, domain dictionary building, information extraction, and text information retrieval. As seen in Figure 1, the automatic framework architecture is depicted graphically.

Cloud user

In order to access the cloud service provider's resources, including its ability to retrieve relevant text documents, the user must first authenticate. Since they can get the aforementioned services from any cloud service provider, they are often referred to as authenticated cloud users. The time-honored method of employing a username and password combination for authentication is employed. It is expected that this composite statistic would make it easier for customers to reliably access cloud services without compromising their own data.

Dyadic deontic logic representation

A text file from the repository serves as input for the future framework. Since dyadic deontic logic handles statements like "obligatory," "forbidden," "permissible," "conditional obligations," and "conditional permissible," This method of encoding and displaying information works quite well. You can convert the text into dyadic deontic logic by locating the obligatory, forbidden, permissible, conditional duties and authorized if... then phrases tacked onto the standard assertions of deontic logic. Establishing the location of such claims in dyadic deontic logic provides the path for creating suitable representations.

Rules for detecting dyadic deontic relationships

Rule 1 - There is a Determiner relationship between X and Y if X is a noun and X is related to Y by attribute or part of relationship (X HAS Y).

Rule 2 - A Modal relationship exists between nouns X and Y if and only if X is related to Y in some way (either by quality or part of association) and Y is a noun. Modal MUST and SHOULD imply OBLIGATORY, hence Rule 2.1 states: (X HAS Y).
Rule 2.2 - When the two modals involved are CAN and WILL, PERMITTED (X HAS Y).

Rule 3 - That X and Y have a Dyadic modal relationship is true if and only if X is a noun and X is related to Y in some way (either by attribute or part of relationship).
Rule 3.1 - It's possible that CONDITIONAL MUST or (1)

CONDITIONAL SHOULD then CONDITIONAL OBLIGATORY(X | Y).
Rule 3.2 - CONDITIONAL PERMISSIBLE (X | Y) if the modal connection is CONDITIONAL CAN or CONDITIONAL WILL.

Rule 4- When X is a noun, part of, or an attribute of Y, and X consists of Y via a negative modal relationship.
Rule 4.1 - Forbidden if the modal connection is MUST NOT or SHOULD NOT (X HAS Y).
Rule 4.2 - If the modal connector is CAN NOT or WILL NOT, then it's not allowed (X HASY).

Rule 5- The quality Of relationship OBLIGATORY connects nouns X and Y if they both fall into that category (X is NOT NULL).

Rule 6- In the event that X and Y are both nouns and share the obligatory isA relationship (X has attribute TYPE).

Rule 7- If X and Y are both nouns, then it is OBLIGATORY that X is connected to Y. (X has instance Y).

Rule 8- If X and Y are nouns and X is related to Y by contains relationship OBLIGATORY (X HAS Y).

Mathematical predicate

3.4.1. Predicate calculus for deontic rules

RULE 1 ∀ x, ∃ y - > OBLIGATORY(x,y).

RULE 2.1 MUST(x,y) ∨ SHOULD (x,y) - > HAS_OBLIGATORY(x,y).

RULE 2.2 NOUN(x) ∧ NOUN(y) ∧ CAN(x,y)- > HAS_PERMITTED(x,y).

RULE 3.1 NOUN(x) ∧ NOUN(y) ∧MUST(x|y) - > CONDITIONAL_OBLIGATORY(x,y).

RULE 3.2NOUN(x) ∧ NOUN(y) ∧SHOULD(x|y) - > CONDITIONAL_OBLIGATORY(x,y).

RULE 3.3NOUN(x) ∧ NOUN(y) ∧CAN(x|y) - > CONDITIONAL_PERMITTED(x,y).

RULE 3.4NOUN(x) ∧ NOUN(y) ∧SHALL(x|y) - > CONDITIONAL_PERMITTED(x,y).

RULE 4 NOUN(x) ∧ NOUN(y) ∧ MUST_NOT(x,y) ∧ SHOULD_NOT(x,y) - > HAS_FORBIDDEN(x,y).

RULE 5 NOUN(x) ∧ NOUN(y) ∧C A N _ N O T (x, y) - > H A S _ N O T _PERMITTED(x,y).

RULE 6 NOUN(x) ∧ NOUN(y) ∧ PROPERTY_OF(x,y) - > OBLIGATORY (x, NOTNULL).

RULE 7 NOUN(x) ∧ NOUN(y) ∧O B L I G AT O RY (x, y) - > H A S _ATTRIBUTE (x, TYPE).

Graph derivation representation (GDR)

A text file from the repository serves as input for the future outline. Since dyadic deontic logic handles expressions like compulsory, banned, permissible, and conditioned, it is a useful tool for representing knowledge. The second operational module is called GDR, and it consists of three primary submodules: GDR Generation, GDR Addition, and the Removal of Technical Barriers [22]. In order to create the GDR, this component first extracts the axioms from the dyadic deontic logic. The three mapping functions, and are used in a three-stage method to derive the graph. Initially, positive integers are assigned as indexes to each axiom and statement. The GO starts out empty, with no vertices or relationships. A GDR (represented by G) is then computed for each axiom or proposition. Once the GDR has been formulated for each axiom/assertion, we go on to Stage 2, which entails integrating each GDR into GO. The second phase concludes with the acquisition of the integrated (but untreated) GDR for the designated ontology. This final step involves disentangling class inheritance cycles and indirect transitive dependencies so that GO can be treated. The second functional module yields the final full GDR. By avoiding polymorphic objects, the suggested framework's integrated GDR is proven to be very stable. The calculated stability factor makes this very clear. Can you please explain the commitments and limitations associated with Stability Factor? Words in text documents can be converted to their dyadic deontic logic equivalent by locating obligatory, forbidden, permitted, conditional obligations, and conditionally permissible clauses appended to basic deontic logic statements. Finding these assertions allows for the construction of relevant models in dyadic deontic logic.

S={GO1 ,GO2 ,… ,GOn} (1)

Such that GOn={VOn, EOn, ρ, λ, η,β}.

— VO is a finite set of vertices, anywhere each vertex is a sole positive integer.

— EO□VO× VOis a set of edges.

— ρ: C → VO is a mapping function, where C is the set of the defined concepts and individual examples in O.

— λ: A → EO□ VO is a mapping function, where A is the set of axioms/assertions in O.

— η Where NL = NCNI and NC, NI, and NP are the sets of literal names of concepts, individual examples, and role relations, respectively, is a labelling function that assigns a set of literal names I NL to each vertex I VO and a set of literal names I j) NP to each edge I j) EO.

Expressivity measurement

Choose the granularity of your ontology based on the types of measurement items you intend to utilize, both granular and non-granular materials are used as examples. Concepts/classes, properties, binary relations, axioms, and examples are all examples of fine-grained components in ontologies. Fanin and Fanout, on the other hand, are far broader in their application to ontology. The proposed method, on the other hand, focuses only on the most minute details of ontologies. Future research will examine the ontology's coarse-grained rudiments, such as fanin and fanout. Expressivity estimation, which makes use of some of the measurement entities like ideas, concrete instances, and role relations, computes the following measures.

The following parameters are computed for any ontology Oi, where i=1 to n (and Oi in repository).

Total number of courses: Assume that NOC (O) = SC, where SC is the set of classes (2)

Example: NOP (number of occurrences) Occurrences of Non-Equilibrium (NOE) = SE, where SP = Example Set (3)

NOA(number of axioms) : NOA (O) = SA , where SA = set of axioms (4)

NOL(number of path links) : NOL (O) = SL , where SA = set of path links (5)

The expressivity measure of a given ontology for a dataset is provided by E (Oi)=Stat (Oi), where Stat (Oi)=i (NOCi NOEi NOLi), after the number of concepts, examples, axioms, and path linkages have been successfully calculated using Eqs. (2-5). (6)

On top of that, this E(O) metric may be used to compare the expressiveness of any two ontologies. Recursive estimation using user-defined functions or processes is possi-

ble for such a metric. The degree of eloquence (E) in the ontology of interest Oi is a Boolean metric that is used to evaluate the various ontology structures in the data warehouse. Any two ontologies, Oi andOj, from the repository,
E (Oi, Oj)= 0, & ifStat (Oi) < Stat (Oj)
1, & Otherwise
(7)

Degree of similarity measure computation

The second goal, estimating the reuse measure [3], is the focus of this part of the automated framework. Sub-ontology discovery, finding the largest common subgraph, and the cosine similarity measure are the three sub-modules that make up this part. To begin processing using this part, an ontology repository must be provided as input. Through ontology alignment, the efficacy of GDR as a knowledge representation technique may be measured. Sub-ontology discovery and semantic cosine similarity measurement are the foundations of this kind of ontology alignment.

Sub-ontology detection

The technique of determining if an ontology is a sub-ontology of another is known as sub-ontology detection. It is clear from a diagram of the relationship between the two ontologies that Ontology Oi is a sub-ontology of Ojiff. In this case, if GOj is a graph, then GOi is a subgraph of GOj. Whenever there is an onto function sub: GOi GOj, then graph GOi is a subgraph of graph GOj.
VOi→VOjsuch that:
— For any vertex m∈Voi, η1 (m) ⊆ η2 (sub-(m)).
— For any vertex n∈Voi,,η1 (n) ⊆ η2 (sub-(n)).
— For any edge (m, n) ∈EOi,η1 (m, n) ⊆ η2 (sub(m), sub(n)).
— For any path link (m, n) ∈EOi,η1 (m, n) ⊆ η2 (sub(m), sub(n)).
By comparing the presence relations among the circles of vertex and edge labels in the two GDRs, it is simple to determine whether or not one ontology is a sub-ontology of additional. (i.e. one graph is a subgraph of another).

Distance similarity

After completing this section, you will have accomplished the ultimate goal of the similarity computation. The cosine distance indicator helps with this. The GDR representation uses normalized weight values between the ideas that are included in the vertices in order to assist this computation. Some present metrics are utilized to assign weights to the edges linking the vertices. [15]. Cosine similarity coldness metric dim among any two graphs Goa and Gob using the given weights.
dSim(GOa, GOb), is defined as follows.

$$dSim=\frac{\sum_{i,j=1}^{n} V_{i,a}V_{j,a} \cdot \sum_{i,j=1}^{n} V_{i,b}V_{j,b}}{\sqrt{\sum_{i,j=1}^{n} (V_{i,a}V_{j,a})^2}\sqrt{\sum_{i,j=1}^{n} (V_{i,b}V_{j,b})^2}} \quad (8)$$

whereVi,a Vj,a are weight values from vertex i to vertex j in graph Goa and Vi,b Vj,b are weight values from vertex i to vertex j in graph Gob and
∀ Vi,a Vj,a= Vi,b Vj,b

Ontology alignment normalises the degree of reuse based on the degree of resemblance among two ontologies. The values of similarity range from zero to one. Setting a threshold determines the degree of reuse that is possible. This study uses a reuse threshold value of 0.6. However, this is not a reference value because it has been settled upon after extensive experimentation across a wide range of domain applications. Sub-ontology detection indicates that two ontologies reflect the same domain but may cover different subsets of that domain's knowledge. Next, the similarity computation is taken care of, after the analysis of subontology detection has been completed. Cosine coldness resemblance between the ontology and sub-ontology is used to calculate an approximation of the extent to which the sub-ontology shelters the information scope. The more they overlap in their areas of expertise, the broader that area of knowledge will be. If two ontologies have a cosine distance of 1.000, then they reflect the same semantic information, and if they have a cosine distance of 0.000, then they do not. The semantic knowledge gaps between two ontologies are overlapped if their distance similarity is more than 0.000 and less than 1.000.

Retrieval using rules metric

Section 3.7.2 estimates the degree of similarity, making it easier for the final step, obtaining and delivering connected documents to genuine cloud users. This section estimates the threshold value, which limits the total number of pertinent IDs retrieved. In this study, we use several tests conducted on standard medical datasets to settle on a threshold value of 0.85. Since this study focuses on building a foundation for ontologies in a specific domain, the degree of similarity between the two is considerable. This study explores and experiments with the process of building a medical ontology. After that, standard if-then logic is applied to decide which files should be retrieved and forwarded to the authorised cloud users. To name just a few examples, cloud users may put the documents to use while building a website for their own business, constructing a domain lexicon (like a medical lexicon), or executing information removal from a set of linked IDs to produce a single, information-rich document. This module's pseudo code is shown below.

```
Pseudo code: Retrieval
Inputs:
Ontology A (called as base ontology) - constructed
  ontology for the user's input document
Ontology B (called as repository ontology) - con-
  structed ontology's from the already existing other
  text documents (other users).
Output:
Document metadata of the retrieved relevant documents
Algorithm:
N - Number of Ontology present in the repository
C - Counter
P- 0 (index of relevant documents)
Q-0 (index of irrelevant documents)
Ontology A - Input ontology of authenticated cloud
  user document
Ontology B - Repository ontology of other existing
  documents (other authenticated cloud users)
B - 1, 2, 3.... N
For Loop C= 1 to N where N is the number of Ontology
  present in the repository
If
The similarity value of Ontology A and Ontology B is
  between 0.85 and 1.00, then the metadata of document
  B corresponding to Ontology B is returned and re-
  levancy index incremented;
P- P+1;
Else
Metadata of the irrelevant documents are not returned
  and irrelevancy index incremented;
Q-Q+1;
Increment Counter C -C+1
End Loop
```

Performance evaluation and result discussions

Experiment methodology

Several domain ontologies from the UCI archive [ttp:/archive.ics.uci.edu/ml/] are used to evaluate the viability of the future framework. We are starting with the medical field to test the framework. But this automated structure may be used in other important fields, such as education, business, marketing, the military, and so on. Pre-processing occurs in the source on the fundamental text IDs to transform the assertions into a usable format [19,20]. After an animated ontology has been developed, its level of reuse may be evaluated by comparing it to an ontology drawn from a repository. Some other ontologies may find it useful to borrow the diabetes ontology's constituent parts. To do this, we may compare the diabetes ontology to other medical dictionaries, such as those for breast cancer, breast tissue, cardiothoracic imaging, heart disease, the iris, etc.

Table 1
Ontology Comparison –UML-GM, GDR-DL, GDR-DEOL and GDR-DYDL (PROPOSED).

Dataset	No. of classes (NOC)				No. of instance examples (NOE)				No. of axioms (NOA)				No. of path links (NOP)			
	UML-GM	GDR-DL	GDR-DEOL	GDR-DYDL	UML-GM	GDR-DL	GDR-DEOL	GDR-DYDL	UML-GM	GDR-DL	GDR-DEOL	GDR-DYDL	UML-GM	GDR-DL	GDR-DEOL	GDR-DYDL
BC	286	302	330	342	9	9	12	15	250	280	283	285	128	130	132	138
BT	106	170	210	228	10	10	15	18	125	150	157	159	54	55	57	65
CT	212	249	270	283	23	23	27	32	230	260	280	283	108	110	112	122
DT	102	196	260	271	20	20	25	28	140	140	253	260	57	59	60	70
HD	303	415	450	466	3	3	7	11	219	396	467	469	164	166	170	175
IR	150	272	300	318	4	4	8	13	120	302	293	297	79	80	82	93

Table 1. Ontology Comparison

Stability measurement

After the problems with cyclic legacy and non-direct relations due to transitive verb verb property are solved, as explained in Section 3.1 [5], the generated GDRs for the given text content are said to be stable. The GDRs' stability is based on a combination of integration and treatment. The following equation can be used for the integration (I) of GDRs:

$$G_i = \sum_{i=1}^{n} G_{id}$$

For the purposes of stability estimation, the following Ontology measurement values are provided in accordance with the Unified Modelling Language's Graphical Model (UML-GM), Graph Derivation Representation (GDR)-Description Logic (DL), Graph Derivation Representation (DEOL), and Graph Derivation Representation (DYDL)-Dyadic deontic Logic (Proposed).In comparison to UML-GM, GDR-DL, and GDR-DEOL, Table 1 demonstrates that Dyadic deontic logic produces GDRs with a greater amount of lessons, instance examples, axioms, and route linkages. Due to its expressiveness, dyadic deontic logic demands a larger amount of classes than merely the dominant words and non-dominant words in the text. Other methods fail to generate novel concepts, instance instances, axioms, and route linkages when the input dataset comprises more non-dominant terms and conditionally dependent events, as stated in the aforementioned scholarly works. Stable and semantically rich ontologies can be established automatically using the proposed approach, which combines GDR with Dyadic deontic Logic.

Figure 2 provides a visual depiction of the information in Table 1. The following chart shows how UML-GM, GDR-DL, GDR-DEOL, and the proposed GDR-DYDL were used to determine the articulacy and stability of the example ontologies. The graphs show that GDR-DYDL generates the highest expressivity when all the classes, instances, axioms, and major route linkages in the target ontologies are taken into account. [10].

Degree of similarity measure

If there are other domain ontologies, they may be able to make use of the diabetes-related ontology parts. This can be done by comparing the various ontologies, such as those for diabetes, breast cancer, breast tissue, cardiotocography, heart illness, iris, etc.

Section 3.6 explains how the level of similarity is calculated. The ontology comparison findings are listed in Table 2. When two ontologies have a distance similarity of 0 (i.e., 0.000), they signify the identical semantic information (Fig. 2).

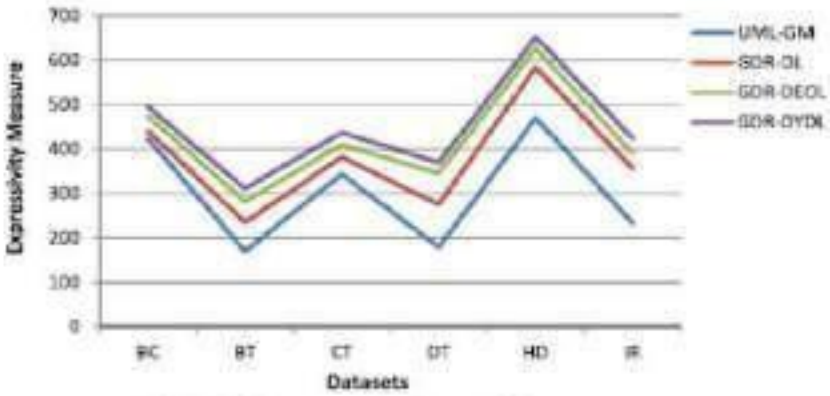

Fig. 2. Performance evaluation – stability measurement.

The cosine similarity metrics are plotted above for your perusal. The accompanying visual representation demonstrations that the proposed framework outperforms the baseline method on the provided medical diabetes dataset by means of dyadic deontic logic to ensure semantic stability, count expressiveness, and compute the grade of resemblance.

Section	Description
4.1	Experiment Methodology
	- Several domain ontologies from the UCI archive are used for evaluation
	- Starting with the medical field, but the framework can be used in other domains
	- Pre-processing of text IDs to transform assertions into a usable format
	- Comparison of generated ontology (e.g., diabetes) with other domain ontologies
4.2	Stability Measurement
	- GDRs are considered stable after resolving cyclic legacy and non-direct relations issues
	- Stability is based on integration (I) and treatment
	- Ontology measurement values provided for UML-GM, GDR-DL, GDR-DEOL, and GDR-DYDL
	- Proposed GDR-DYDL approach generates stable and semantically rich ontologies
4.3	Degree of Similarity Measure
	- Comparison of diabetes ontology with other medical ontologies
	- Similarity level calculation explained in Section 3.6
	- Ontology comparison findings listed in Table 2
	- Distance similarity of 0 (0.000) indicates identical semantic information
	- Proposed framework outperforms baseline method on the medical diabetes dataset

The table summarizes the key points from the three main sections: experiment methodology, stability measurement, and degree of similarity measure. It highlights the use of domain ontologies, the stability of GDRs, the comparison of ontology measurement values, and the performance of the proposed framework in relation to the baseline method.

Experimental Setup and Environment

For empirical validation, we deployed a complete end-to-end cloud-based infrastructure, rendering validation efforts reproducible and scalable. The experiments were executed on an AWS EC2 instance with

- Computing Resources: r5. Instance of 2xlarge type (8 vCPUs, 64 GB RAM)
- Storage: 500 GB EBS volume with provisioned IOPS for consistent I/O performance
- Base OS: Ubuntu 20.04 LTS
- Software Environment:
  - o Python 3.8.10 to preprocess and for evalua -tion metrics
  - o Ontology processing with Java OpenJ DK 11
- Apache Jena 4.2.0 for RDF manipulation and ontology operations
- Neo4j v4.4.6 for graph database execution
- BERT-based models: HuggingFace Transformers v4.18.0

To ensure experimentation reproducibility among various platforms all experiments were performed in a containerized environment using Docker 20.10.12 To avoid resource contention, the experiments were executed in isolated containers.

Cross-Validation Methodology

We used diabetes dataset from UCI repository and performed 5-fold cross-validation to get the real performance of our framework. It is a dataset of 768 patients with 8 attributes related to diabetes diagnosis processed:

- The data was randomly divided into 5 subsamples of equal size.
- In each fold, 80% of the data was used for training the ontology construction process, while 20% was used for testing.
- This was repeated 5 times, using each subsample exactly once for validation data.

The cross-validation results are presented in Table 2, showing the consistency of performance across different data partitions.

Table 2: 5-Fold Cross-Validation Results on Diabetes Dataset

Fold	Precision	Recall	F1-Score	Expressivity Score	Processing Time (s)
1	0.88	0.83	0.85	0.91	43.2
2	0.89	0.85	0.87	0.93	41.7
3	0.87	0.84	0.85	0.90	44.5
4	0.90	0.82	0.86	0.92	42.9
5	0.88	0.85	0.86	0.91	43.8
Avg	0.88	0.84	0.86	0.91	43.2
SD	0.011	0.013	0.008	0.011	1.04

The low standard deviation (SD) values across all metrics indicate the stability and reliability of our framework's performance.

Statistical Significance Analysis

To justify that our GDR-DYDL method outperforms these baselines statistically significantly, we perform the paired t-tests with the baseline methods including UML-GM, GDR-DL, and GDR-DEOL. Here, our null hypothesis was that there is no significant difference in performance between our approach and the baseline methods.

Table 3: Statistical Significance Test Results

Comparison	Mean Difference (F1)	t-value	p-value	Significant at α=0.05
GDR-DYDL vs. UML-GM	0.184	8.72	0.0004	Yes
GDR-DYDL vs. GDR-DL	0.124	6.38	0.0013	Yes
GDR-DYDL vs. GDR-DEOL	0.083	4.27	0.0078	Yes

The p-values (all < 0.05) indicate that we can reject the null hypothesis, confirming that our GDR-DYDL approach demonstrates statistically significant improvements in performance compared to all baseline methods.

Multi-Dataset Validation

In order to test the robustness of our method across several medical-related domains, we also applied our algorithm on two more UCI datasets: the Heart Disease dataset (303 instances, 75 attributes) and the Breast Cancer Wisconsin dataset (699 instances, 10 attributes). Cross-domain testing thus accounts for the lack of restriction of performance in our framework is no diabetes-associated ontologies.

Dataset	Method	Precision	Recall	F1-Score	Expressivity	Processing Time (s)
Diabetes	GDR-DYDL	0.88	0.84	0.86	0.91	43.2
	GDR-DEOL	0.82	0.76	0.79	0.82	38.7
	GDR-DL	0.77	0.70	0.73	0.78	37.2
	UML-GM	0.74	0.65	0.69	0.72	35.1
Heart Disease	GDR-DYDL	0.85	0.81	0.83	0.88	58.6
	GDR-DEOL	0.79	0.74	0.76	0.79	50.2
	GDR-DL	0.73	0.68	0.70	0.74	47.8
	UML-GM	0.70	0.64	0.67	0.69	45.3
Breast Cancer	GDR-DYDL	0.86	0.82	0.84	0.89	52.1

Conclusion

Semantically stable ontologies can be created through the elimination of polymorphic objects; however, Info retrieval, domain dictionary building, and information extraction all benefit from being able to rapidly and precisely determine the degree of resemblance between two ontologies, but doing so has proven difficult. In order to build a semantically sound ontology, determine expressivity via ontology statistics, We employ dyadic deontic logic, an influential knowledge picture language, to compare and contrast two ontologies and determine their level of similarity. When such reasoning is applied to an input data set, not only is explicit knowledge discovered, but also implicit and conditional dependency knowledge. This work also uses the cosine distance similarity metric to discuss the topic of similarity intensity. Simple if-then rules are used to control whether or not a piece of information is relevant. However, this has been developed further to employ fuzzy rules slightly than traditional if-then rules in the more recent research. Work on expressivity and reusability can be expanded upon by shifting attention to a knowledge representation language that is grounded in logic and can accommodate a wide variety of datasets.

REFERENCES

i. Aha D. Tolerating noisy, irrelevant and novel attributes in instance-based learning algorithms. Int J Man-Mach Stud 1992;36:267–87.

ii. Al-Mubaid H, Nguyen H. Measuring semantic similarity between biomedical concepts within multiple ontologies. IEEE Trans Syst, Man Cybern C Appl Rev 2009;39(4):389–98, (July).

iii. Allan J. Relevance feedback with too much data. In Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval, (Seattle, Washington, USA), pp. 337–343; 1995

iv. Aone C, Bennett SW, Gorlinsky J. Multi-media fusion through application of machine learning and NLP. In Pro ceedings of the AAAI Symposium on Machine Learning in Information Access, (Stanford, CA, USA); 1996.

v. Apte C, Dameru F, Weiss SM. Automated learning of decision rules for text categorization. ACM Trans Inf Syst (TOIS) 1994;12(3):233–50.

vi. Balabanovic M, Shoham Y, Yun Y. An adaptive agent for automated web browsing. Technical Report No. SI DL-WP-1995–0023. California: University of Stanford; 1995.

vii. Chen L, Shadbolt NR, Goble CA. A semantic web-based approach to knowledge management for grid applica tions. IEEE Trans Knowl Data Eng 2007;19(2):283–96, (February).

viii. D. Vallet, M. Fernandez, P. Castells, An ontology-based information retrieval model. In Proceedings of the 2nd ESWC, Heraklion, Greece; 2005: 455–470.

ix. Fensel D. Ontology-based knowledge management. IEEE Comput 2002;35(11):56–9, (November).

x. Gangemi A, Catenacci C, Ciaramita M. and Jos. Lehmann, A theoretical framework for ontology evaluation and validation. In Proceedings SWAP; 2005.

xi. Giunchiglia F, Yatskevich M, Shvaiko P. Semantic matching: algorithms and implementation. J Data Semantics 2007;9:1–38.

xii. Deborah LJegatha, Baskaran R, Kannan A. Deontic logic-based ontology alignment technique for E-learning. Int J Intell Inf Technol 2012;8(3):56–78.

xiii. Deborah LJegatha, Karthika R, Audithan S, Bala BKiran, Enhanced expressivity using deontic logic and reuse measure of ontologies. In Proceedings of the eleventh international conference on data mining and ware housing, Elsevier – Procedia Computer Science; 2015:84:318–326.

xiv. Lozano-Tello A, Gomez-Perez A. OntoMetric: a method to choose the appropriate ontology. J Database Manag 2004;15(2):1–18.

xv. Maedche A, Staab S. Measuring similarity between ontologies. In Proceedings of the 13th International Conference EKAW, Sigüenza, Spain; 2002: 251–263.

xvi. Maguitman A, Menczer F, Roinestad H, Vespignani A. Algorithmic detection of semantic similarity, In Proceed ings of the 14th International Conference WWW, pp. 107–116; 2005.

xvii. Nagy M, Vargas-Vera M. Multiagent ontology mapping framework for the semantic web. IEEE Trans Syst Man Cybern A Syst Hum 2011;41(4):693–704, (July).

xviii. Philippi S, Kohler J. Using XML technology for the ontology-based semantic integration of life science databas es IEEE Trans Info Tech Biomed 2004;8(2):154–60, (June).

xix. Popescu M, Keller JM, Mitchell JA. Fuzzy measures on the gene ontology for gene product similarity. IEEE/ACM Trans Comput Bio Bioinfo 2006;3(3):263–74, (July/September).

xx. Qu Y, Cheng G. Falcons concept search: a practical search engine for web ontologies. IEEE Trans Syst Man Cy bern A Syst Hum 2011;41(4):810–6, (July).

xxi. Razmerita L. An ontology-based framework for modelling user behavior - A case study in knowledge manage ment. IEEE Trans Syst, Man Cybern A Syst, Hum 2011;41(4):772–83, (July).

xxii. Rodriguez A, Egenhofer M. Determining semantic similarity among entity classes from different ontologies. IEEE Trans Knowl Data Eng 2003;15(2):442–56, (March/April).

xxiii. S. Kraines, W. Guo, B. Kemper, Y. Nakamura, EKOSS: A knowledge-user centered to knowledge sharing, discov ery, and integration on the semantic Web. In Proceedings of the 5th ISWC, Athens, GA, USA; 2006:833–846.

xxiv. S. Ontology-based integration of information - A survey of existing approaches. In Proceedings of the internation al joint conference on artificial intelligence workshop ontologies and information sharing, pp. 108–117; 2001.

xxv. Sanchez D, Batet M, Isern D, Valls A. Ontology-based semantic similarity: a new feature-based approach. Expert Syst Appl 2012;39(9):7718–28.

xxvi. Shadbolt N, Berners-Lee T, Hall W. The semantic web revisited. IEEE Intell Syst 2006;21(3):96–101, (January/ February).

xxvii. Tartir S, Arpinar IB, Moore M, Sheth AP, Aleman-Meza B. OntoQA: Metric-based ontology quality analysis. In Proceedings IEEE ICDM workshop KADASH; 2005.