A Framework for Enhanced Multi-Concept Based MaCOnto Ontology Evolution

Godwin Sunday Oloruntoba[1], Enesi Femi Aminu[2], Sulaimon Adebayo Bashir[3]
[1.2.3]Department of Computer Science, Federal University of Technology, Minna Nigeria.
oloruntoba.godwin@fccpc.gov.ng[1], enesifa@futminna.edu.ng[2], bashirsulaimon@futminna.edu.ng[3]

***Abstract*—**The discovery of ontology as semantic knowledge representation technique in solving the problems of unstructured data, concept mismatch is no doubt promising. However, the process of developing ontology manually based on the top down design approach is not just tedious and time consuming, but makes the ontology files static. Attempts have been made to solve this limitation by literature but not without some gaps. MaCOnto, a maize crop ontology attempted to solve the problem of ontology dynamism, but only one concept can autonomously evolved. Therefore, this research proposes a robust framework that can enable multiple related concepts of an existing ontology to evolve autonomously, a case study of MaCOnto in this research. The proposed design leverages on machine learning and deep learning techniques. The framework facilitates the autonomous evolution of the MaCOnto ontology through effective concept extraction and relationship mapping from diverse data sources, such as Wikipedia. Additionally, the research utilized the Term Frequency-Inverse Document Frequency (TF-IDF) for extracting domain-related concepts and Word2Vec algorithms for generating contextual word embeddings. The findings emphasize the significance of developing adaptive ontologies capable of evolving with dynamic knowledge domains, thus improving knowledge representation and reasoning in agricultural applications. Results indicate that the enhanced MaCOnto framework successfully achieved an average performance success rate of 83.5% across various competency questions, demonstrating its effectiveness in autonomously encoding multiple concepts and improving overall ontology functionality.

Keywords—MaCOnto, Maize Crop Ontology, Multi-Concept, Ontology Evolution, Word2Vec.

I. INTRODUCTION

The term "ontology" in Artificial Intelligence (AI) is often confused due to its philosophical origins, which pertain to the study of existence, and its overlap with epistemology, which deals with knowledge (Gruber, 1993). In AI, the term ontology formally represents key concepts, relationships, and vocabulary within a specific domain. This structured representation enables effective communication, data integration, and knowledge sharing among systems and agents (Hawley, 2019; Smith, 2012). Ontologies serve as frameworks for organizing knowledge and are essential for interconnections in knowledge management systems. They enable automated reasoning and enhance data quality by providing a consistent understanding of information across diverse domain (Kotis et al., 2020). Despite their advantages, ontologies face challenges such as structural incompatibility with new data sources due to strict formatting rules.

Moreover, Creating a shared ontology helps agents or systems understand and share information without being held back by different terms or data formats. This leads to better teamwork and allows for making decisions based on the combined knowledge in the ontology (Hassan et al., 2021). The Semantic Web depends on formal ontologies to organize data, making it easier for machines to understand information. However, the increasing number of ontologies has made it necessary to find better ways to develop them (Dou et al., 2015). Building ontologies manually can be slow and complicated, leading to issues with gathering knowledge, time management, and integrating different systems (Osman et al., 2022). To address these challenges, ontology learning has emerged as a solution. This process involves automatically or semiautomatically extracting important concepts and relationships from data to create ontologies. This approach helps overcome the difficulties of manual knowledge gathering and supports the goals of the Semantic Web (Hassan et al., 2021).

Additionally, with the large amounts of data generated on daily bases, the focus has moved toward supporting data-driven ontology development. This involves creating ontology specifications based on the type of data and linking these specifications with knowledge from domain experts (Lee et al., 2016). This approach helps in understanding the data, evaluating its quality, and addressing differences among various data sources (Tudorache, 2020). Khadir et al., (2021) highlight that ontologies are fundamental to the Semantic Web, serving as essential knowledge bases for various artificial intelligence applications. However, the manual construction of ontologies is a tedious and labor-intensive process, leading researchers to explore an automated approach known as ontology learning. This technique automates several tasks in the ontology development process, aiming to simplify and streamline ontology creation (Kotis et al., 2020). By utilizing ontology learning, the time and repetitive effort involved in manual construction can be significantly reduced, as these automated methods extract relevant concepts, relationships, and other elements from diverse data sources, resulting in more efficient and scalable ontology development. The significance of ontology learning lies in its potential to enhance the adoption and use of ontologies in AI applications by making the construction process more accessible and less resource-intensive, thereby facilitating the integration of semantic technologies and knowledge-driven systems (Khadir et al., 2021). Overall, while ontology engineering including development, mapping, and integration has expanded significantly in areas like semantic information systems and natural language processing, challenges remain regarding the scalability of domain ontologies (Kotis et al., 2020).

Lastly, to address the scalability limitations of domain ontologies, Aminu et al., (2022) developed a novel application-based ontology for maize crops called MaCOnto, which incorporates knowledge about fertilizers, soils, irrigation, and climatic conditions. The literature also identifies other domain and application-based ontologies in fields such as medicine, finance, and education (Bunnell et al., 2021; Larentis et al., 2021) . Given the robustness of MaCOnto and the goal of enabling it to evolve autonomously, this research proposed an enhanced Multi-concept based ontology which involved learning multiple concepts to establish relationships within an existing ontology.

A. Research Problem

The application of ontology development as a knowledge representation model has been steadily expanding, but currently no standard automated tools are available for ontology engineers, making manual development a common approach to ensure consistency (Fawei et al., 2019). However, enabling the autonomous evolution of ontologies poses a challenge, as ontology files should not remain static. To address this issue, Aminu et al., (2022) developed the Maize Crop ontology based knowledge (MaCOnto) and also introduced the MaCOnto Evolution Algorithm, but the designed framework struggled to manage multiple relevant concepts from users' competency questions (CQ's). Similarly, Pietranik & Kozierkiewicz, (2023) sought to tackle the broader problem of ontology evolution and scalability by creating a framework for ontology evolution and alignment; however, this framework lacks autonomous capabilities.

B. Research Goal

This study aims to enhance the existing MaCOnto Ontology Evolution system developed by Aminu et al., (2022), by introducing a multi-concept-based capability using Machine learning and deep learning techniques.

C. Contribution to Knowledge.

This research is significant as it addresses the challenges of manual ontology engineering, which, while still viable, is labor-intensive due to the absence of standard automated tools and fails to support autonomous ontology evolution since ontologies should not remain static. (Aminu et al., 2022) developed the MaCOnto Evolution Algorithm to tackle these issues but found limitations in handling multiple relevant concepts from user competency questions, indicating a need for more comprehensive solutions. Similarly, (Pietranik & Kozierkiewicz, 2023) aimed to improve ontology evolution scalability through a framework for ontology evolution and alignment, but this framework also lacks autonomy. Thus, developing effective techniques for autonomous ontology evolution is crucial for creating adaptive ontologies that can evolve with the dynamic nature of knowledge domains and increasing data volumes, ultimately enhancing knowledge representation and reasoning in agricultural applications like maize.

II. RELATED WORKS

This section introduces the literature on ontology development and evolution by highlighting the significant advancements and persistent challenges in the field of Ontology and Ontology Evolution, particularly regarding the need for automated solutions and scalability. Aminu et al., (2022) introduced the MaCOnto ontology for maize crops, addressing issues related to unstructured data and concept mismatches in agriculture. The researcher's work emphasized the importance of autonomous growth in ontologies, although it also revealed limitations in handling multiple relevant concepts, indicating a gap that requires further exploration. Similarly, (Safyan et al., 2019) proposed a framework for ontology evolution that adapts to new knowledge through personalized activity recognition models. While this framework demonstrates dynamic adaptation, it does not fully address scalability or computational efficiency, suggesting additional areas for research.

In the realm of automatic ontology construction, (Al-Aswadi et al., 2020) noted the challenges of manual methods and reviewed various approaches to automate ontology creation from text, yet they failed to provide a comprehensive evaluation or identify specific gaps in applying deep learning techniques. (Yang et al., 2020) tackled the fragmentation of systems engineering standards by proposing an ontology learning methodology that employs natural language processing to extract relevant terminologies and concepts from existing documents, although their study lacked a thorough evaluation of the proposed methodology.

Cardoso et al., (2020) highlighted the need for maintaining historical knowledge within ontologies to improve annotation quality across different versions, introducing a Historical Knowledge Graph approach that addresses this issue effectively but lacks detailed evaluation. (Babaei Giglou et al., 2023) explored the potential of Large Language Models (LLMs) for ontology learning, finding that while foundational models may not suffice for complex reasoning tasks, fine-tuning could enhance their utility in knowledge acquisition.

Furthermore, Urdaneta-Ponte et al., (2021) developed a hybrid recommendation system leveraging ontologies to assist professionals in lifelong learning, demonstrating notable improvements over previous systems but lacking extensive evaluation across diverse contexts. (Navarro-Almanza et al., 2020) proposed a deep learning-based methodology for automatic ontology construction with minimal human intervention; however, they did not assess its scalability or efficiency comprehensively. Li et al. (2020) also introduced a framework called BOLT to support behavioral research by addressing knowledge inaccessibility through an ontology-based search engine, although further evaluation across broader domains is needed. Lastly, Li & Chen (2020) presented an ontology-driven approach for classifying security requirements that outperformed existing methods but similarly lacked comprehensive assessments across various application domains.

III. RESEARCH METHODOLOGY

This section introduces the research methodological approach used to develop the Multi-Concept MaConto Ontology evolution algorithm, along with the Ontology Evolution Conceptual Framework. It includes a system flow chart, use case instances, and the algorithm itself. Additionally, the performance evaluation of the adopted Multi-Concept machine learning and deep learning-based algorithm is presented, assessing the robustness of the evolved MaCOnto ontology through competency questions (CQs) related to the domain of Maize Crop Ontology.

A. Analysis of Existing System (MaConto)

The MaCOnto ontology, as developed by *(Aminu et al., 2022),* is grounded in the Maize Crop Knowledge Framework (MCKF) and encompasses four primary knowledge domains: MaizeCrop (Mo), Soils (So), Fertilizers (Fo), and Irrigation (Io). In addition to these domains, the ontology incorporates supportive knowledge (SKo) that includes Climatic Conditions (such as wind speed, humidity, sunlight, and temperature), Farm Implements, Pest/Insects, and Diseases. This ontology serves as a secondary data source in this study for the collection and extraction of domain concepts, facilitating the expansion of its knowledge base through machine learning and deep learning ontology evolution techniques.

Based on the comprehensive review conducted on the existing study the following limitations are identified.

- Reliance solely on a statistical approach for feature extraction, specifically using TF-IDF
- The absence of deep learning methods for automatic feature extraction
- The MaCOnto ontology evolution process, which is restricted to evolving by adding only one concept at a time
- Dependency on a predefined ontology, such as WordNet, for concept extraction; and
- limitations in relation encoding between objects and instances, which are confined to hypernyms, meronyms, hyponyms, and holonyms relationships.

Algorithm 1 shows the existing MaCOnto evolution algorithmic expression.

Algorithm 1: Existing MacOnto Ontology Evolution

Input: Q
Output: Result, MaCOnto Autonomous Update
Parameters: Query String (Q); WordNet; MaCOntoTerms (T); POSparser: CandidateTerms (C); counter (i); numbers of terms in C (n); hypernyms (hyp); holonymy (hoy); Meronym (mer); domain (dom); object property (obProp); individual (indiv)
Procedure:
execute OntEvolutionThreads;
input Q
qPOS=POSparser(Q, true); //preprocess to remove stop words and tokenize
C = qPOS;
initialize T;
do
if C ∈ T then // if C is found among MaCOnto's Terms
Output Result // the appropriate C along with synonyms terms and relation
elseIf C ∉ T. then //if C is not found in MaCOnto (MaCOnto starts to evolve)
learn WordNet (C);

```
check SemanticRelation(hyp; hoy; mer);
encode OntStatement(dom; range; subclass; obProp);
elseIf find(WordNet; classes) then
axiom; classes = getRelatedAxiom(WordNet);
end
elseIf find (WordNet; indiv) then
axiom; indiv = getRelatedAxiom(WordNet);
end
update MaCOnto;
else Print "out of subject granularity" end
end
end
while (i = n)
return MaCOnto
```

Based on the algorithm its clearly shown that the existing work primary using Wordnet as the core knowledge based and statistical approach to Evolved on the existing MacOnto file.

B. Multi-Concept MaCOnto Framework

The proposed Framework is developed to enhance the existing MaConto Framework by introducing the ability to evolve on multiple concept dynamically, hence Mulit-Concept Based MacOnto.

However, the Multi-Concept MacOnto Framework, depicted in Figure 1, outlines the detailed technological approach employed to achieve the proposed Multi-Concept MaCOnto Ontology Evolution. As illustrated in Figure 1, the Wikipedia data source is utilized for collecting raw text data, which is subsequently filtered based on concepts extracted from the existing MaCOnto knowledge base using Term Frequency-Inverse Document Frequency (TF-IDF). Then the next step involves concept extraction through document tokenization and the generation of word embeddings using a Word2Vec deep shallow neural network architecture. These embeddings facilitate the extraction of synonymous concepts that are contextually similar to the original terms, collectively forming the Multi-concepts. Additionally, relationships (predicates) between subjects and objects within the raw text corpus are extracted via machine learning-based Part-of-speech (POS) pattern extraction. The Subject-Predicate-Object (SPO) patterns derived from the raw text are then mapped to Class-Relationship and Instance (CPI) patterns within the ontology's relational database representation.

The framework employs an Operational Derived Hierarchy (ODH) Technique to generate a concept hierarchy based on is-a relationships. Following this, a relational data structure comprising Subject, Predicate, and Object is established and utilized to construct a knowledge graph, which also aids in enhancing the existing MaCOnto ontology through its evolution. Ultimately, there is a continuous synchronization between the existing ontology and the Ontology Concept SPO structure database, ensuring that both remain up-to-date and aligned.

1) Data Source

This study considers two major primary data sources for extracting essential candidate terms critical to the evolution of the Maize Crop Ontology: Wikipedia and the existing MaCOnto Ontology. Ontologies serve as formal models that define concepts, properties, and relationships within a domain, facilitating unambiguous interpretation of data from diverse sources. The MaCOnto is an OWL and FOL-based application ontology for maize crops, from which essential concepts can be derived. Wikipedia, with over one million entries, provides a rich repository of concepts and definitions for ontology development. The

URIs of Wikipedia articles can function as identifiers for ontology classes, while the article text offers definitions and descriptions of various domain terms. This study will leverage the existing MaCOnto to collect documents from the Wikipedia data source, which will subsequently be used to evolve and expand the knowledge base of MaCOnto through ontology engineering.

..

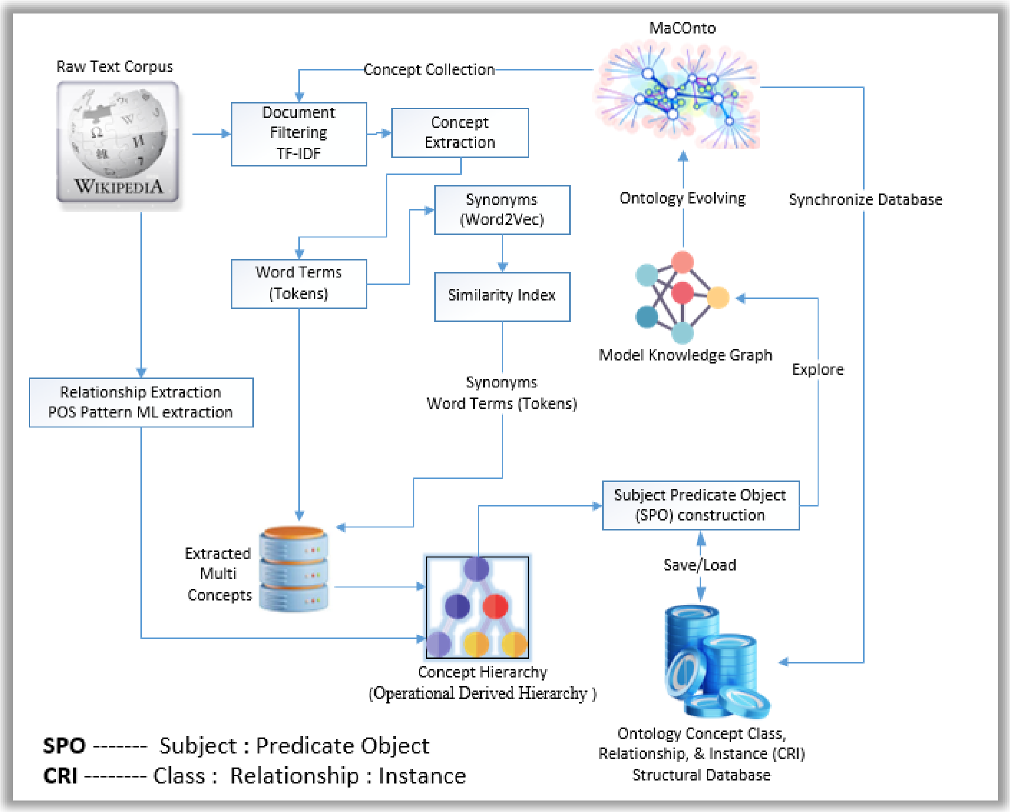

Figure 1: Proposed Multi-Agent MaCOnto Framework

2) Text Preprocessing

To derive meaningful patterns and perform computational activities from raw text data, it is essential to preprocess the data and transform it into actionable insights. This study will conduct various preprocessing operations on the raw text data collected from Wikipedia and the MaCOnto knowledge base, which will include applying TF-IDF for feature extraction, generating word embedding using Word2Vec, performing Part of Speech tagging, constructing a concept hierarchy, extracting Subject-Predicate-Object (SPO) relationships, and calculating similarity indices.

3) TF-IDF

The Term Frequency-Inverse Document Frequency (TF-IDF) denotes a statistical measure employed to assess the significance of a word within a document or corpus. In this study, essential terms are extracted from the existing MaCOnto and utilized to filter documents that exhibit a high level of relevance to the concepts of Maize Ontology. The equation below mathematically illustrates this process:

$TF - IDF\ (t,d) = TF\ (t,d)\ \times IDF(t)$……………Eq(1)

Where:

TF(t,d): denote the term frequency of term ‘t’ in the document ‘d’

IDF(t): is the inverse document frequency of term ‘t’, which is calculated as:

$IDF(t) = \log\left(N/df(t)\right) + 1$. Where N is the total number of documents and df(t) is the document frequency of the term “t”

4) Word Embeddings (Word-2-Vec)

Word embeddings, such as Word2Vec, represent words as numerical vectors that encapsulate both semantic and syntactic relationships among them. The Word2Vec model can be trained on a text corpus to generate these vectors, which are subsequently used to identify related concepts and their interrelations. The word embeddings approach is employed in this study to find words with contextual similarity, specifically focusing on synonyms that can be interchangeably used based on their context.

5) Part of Speech Tagging

Part-of-speech (POS) tagging involves assigning grammatical tags (such as nouns, verbs, or adjectives) to each word in a text. This information is essential for identifying the types of concepts such as entities, attributes, and relationships present in the text, which is critical for constructing the ontology structure. In this study, the SpaCy and NLTK libraries will be utilized to identify and tag each word in a sentence with its appropriate part of speech.

6) Concept Hierarcy

The concept hierarchy represents the taxonomic structure of the ontology; wherein more general concepts are connected to their more specific sub-concepts. This hierarchy can be established by analyzing the semantic relationships between terms in the text corpus, particularly focusing on hypernym-hyponym (is-a) relationships. In this study, the Subject-Predicate-Object (SPO) extraction approach will be employed, as detailed in the following subsection.

7) Subject Predicate Object (SPO) Extraction

SPO extraction involves identifying subject, predicate, and object triples from raw text, which can then be used to populate the ontology with Classes, Relationships, and Instances (CRI). This process employs natural language processing techniques and pattern matching. The SpaCy pre-trained model will be utilized to perform dependency parsing and extract syntactic information, including subjects (nouns, pronouns), predicates (verbs, verb phrases), and objects (nouns, pronouns). This pattern facilitates the creation of a structured table or database that can be synchronized with the MaCOnto ontology.

C. *Use case diagram Design*

Figure 2 illustrates the proposed system's use case instances as experienced by end users. According to the figure, the system is directly utilized by a single actor (the end user). The following use case instances are conducted by the actor:

i. Initiate System: The user can launch the application.
ii. Enter Query: The end user can input a query via the provided link.
iii. Concept Extraction (Enter Query): Upon query entry, the system performs concept extraction on the submitted input.
iv. Documents Ranking (Enter Query): This indirect case involves utilizing the user input to rank documents from the MaCOnto relational database storage.
v. View Response: The user can view the application's response based on their query.
vi. Response (View Response): The responses displayed to the end user may include valid responses, invalid queries, or queries not found.
vii. Invalid Query (View Response): When an invalid query is encountered, the system engages in self-learning of new concepts (Multi-Concept), a process known as Ontology Evolution.
viii.

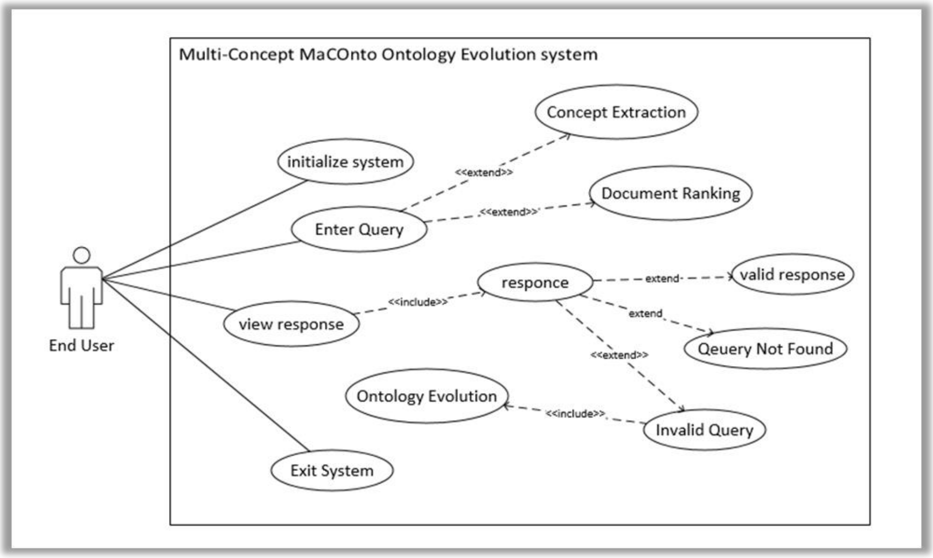

Figure 2: System Use Case

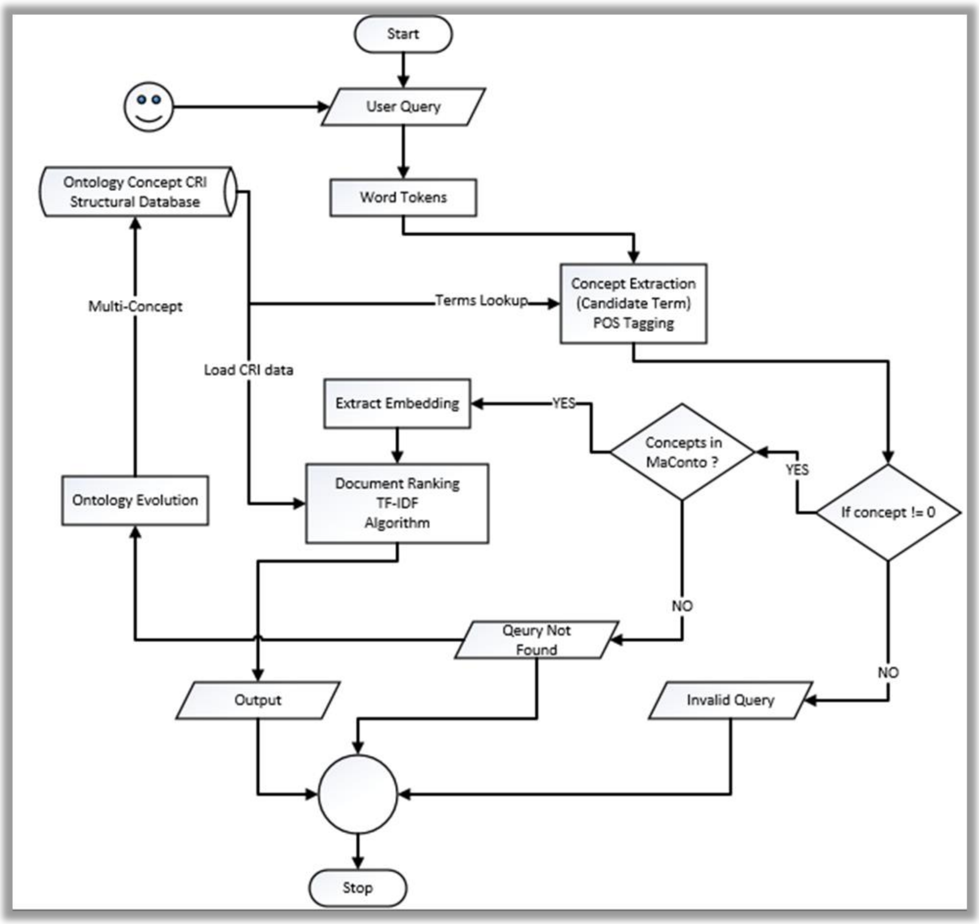

Figure 3: System Flowchart Diagram

The flowchart diagram illustrates the sequential steps of data flow within the application framework. Figure 3 depicts the systematic progression of data from user input to application responses. According to the figure, the query string is collected from the end user and subsequently tokenized into individual words. These tokens are analyzed using a Part-of-Speech parser, which tags each word as either a subject, predicate, or object. If the analyzed text contains more than one concept and these concepts are found within the MaCOnto ontology, further processing can be performed on the document, including document ranking using the TF-IDF algorithm. The documents are sourced from the Ontology Concept SPO structural database.

Furthermore, if the concepts extracted from the user query do not match any of the concepts in the MaCOnto ontology, the system will output "Query Not Found," prompting the ontology to evolve through the proposed Multi-Concept Ontology Evolution process. Lastly, if no concepts are found in the database, the end user will receive an "Invalid Query" response.

D. *Propose System Algorithm*

Algorithm 2: Proposed Multi-Concept Ontology Evolution Concept Framework

Input `Q`.
If `Q ∉ MaCOnto`:
1. Fetch `Corpus` from Wikipedia.
2. Filter `Corpus` using TF-IDF to get `Filtered_Documents`.
3. Tokenize `Filtered_Documents` to get `T`.
4. Train Word2Vec on `T` to get `Word_Embeddings`.
5. Compute `Cosine_Similarity` to find similar words `S`.
6. Merge `T` and `S` into `Multi_Concept_Document`.
7. Construct `Hierarchy` from `Multi_Concept_Document`.
8. Extract relationships `R` using POS pattern matching (Subjects `S`, Objects `O`, Predicates `P`).
9. Save `R` in `CRI_Database`. // C: Class R: Relationship I: Instance database
10. Create `Knowledge_Graph` from `R`.
11. Update `Ontology` with `Knowledge_Graph`.
12. Synchronize `Ontology` with `SOP_Database`.
13. Update `MaCOnto` with the updated `Ontology`.

According to Algorithm 1, steps 4 through 11 are crucial to the ontology evolution process. Step 4 calculates term frequency-inverse document frequency (TF-IDF) to facilitate document ranking, while steps 5 and 6 involve extracting all synonyms associated with each term using word embedding's and cosine similarity computations. Steps 6, 7, and 8 focus on extracting all possible concepts (classes or instances) from the document, classifying these concepts into a hierarchy, and identifying relationships between sentences using Part of Speech Matcher machine learning techniques. Finally, this structured pattern is utilized to model the Class Relation and Instance (CRI), which will be further employed to autonomously integrate new knowledge into the existing MaCOnto ontology and synchronize the Subject-Predicate-Object table with the Ontology Class Relationship and Instance.

IV. FRAMEWORK IMPLEMENTATION

The implementation of the MaCOnto Evolution algorithm necessitates a comprehensive environmental setup, including the programming language, development environment, and essential dependencies required for the successful development of a self-learning, multi-concept-based MaCOnto ontology. The development environment will consist of Jupyter Notebook and Visual Studio Code, with Python (Version 3.9x) selected as the programming language due to its extensive library support for AI and automated system development. Key Python modules such as Scikit-Learn, TensorFlow, SpaCy, Seaborn, and Selenium will be employed to implement the proposed Autonomous Multi-Concept Evolution System. The entire development process will be fully implemented within the Jupyter Notebook environment.

V. EXPERIMENTAL RESULT & EVALUATION

The framework performance is evaluated using competency questions that are not manually encoded in the existing MaConto Ontology File. The Multi-Concept Ontology will autonomously encode new concept using the proposed framework concept. Table 1 show the performance and expected result of the proposed Multi-Concept MaCOnto ontology.

Table 1: Performance Evaluation Table.

S. No	CQ's	Candidate Terms	Proposed Multi-Concept MaCOnto Evolution Framework (New Concepts)	Expected Concept	Generated Concept	Success Rate/Accuracy
1	What are the effects of fertilizers on maize yield?	Fertilizers, Maize, Yield	Nutrient Uptake Efficiency, Fertilizer Application Methods, Soil Nutrient Status	Fertilizer Impact on Yield	FertilizerEffectOnYield	85%
2	How does irrigation affect maize growth?	Irrigation, Growth, Maize	Irrigation Techniques, Water Use Efficiency, Soil Moisture Retention	Irrigation Impact on Growth	IrrigationEffectOnGrowth	80%
3	What diseases affect maize crops?	Diseases, Maize	Disease Resistance Mechanisms, Pathogen Types, Symptoms and Diagnosis	Disease Classification	MaizeDiseaseTypes	90%
4	What are the soil requirements for maize cultivation?	Soil Requirements, Maize	Soil pH Levels, Nutrient Composition, Soil Texture and Structure	Soil Requirements for Maize	MaizeSoilRequirements	75%
5	How do pests influence maize production?	Pests, Production, Maize	Pest Management Strategies, Economic Threshold Levels, Pest Life Cycles	Pest Influence on Production	PestImpactOnProduction	88%
6	What is the relationship between maize and climate?	Climate, Maize	Climate Adaptation Strategies, Temperature Sensitivity, Precipitation Patterns	Climate Impact on Maize	ClimateEffectOnMaize	82%
7	How do organic compounds affect soil health?	Organic Compounds, Soil	Organic Matter Decomposition, Soil Microbial Activity, Carbon Sequestration Potential	Organic Impact on Soil Health	OrganicCompoundSoilHealth	78%
8	What is the role of maize in food security?	Food Security, Maize	Maize Production Statistics, Food Distribution Networks, Nutritional Value of Maize	Role of Maize in Food Security	MaizeFoodSecurityRole	85%
9	How does crop rotation	Crop Rotation, Yield	Crop Diversity Benefits, Soil Health Improvement, Pest and Disease Suppression	Crop Rotation Impact on Yield	CropRotationEffectOnYield	80%

	affect maize yield?					
10	What are the economic impacts of maize farming?	Economics, Farming, Maize	Cost-Benefit Analysis of Farming Practices, Market Demand Trends, Subsidy Effects on Production	Economic Impact of Maize Farming	EconomicImpactOfMaizeFarming	77%

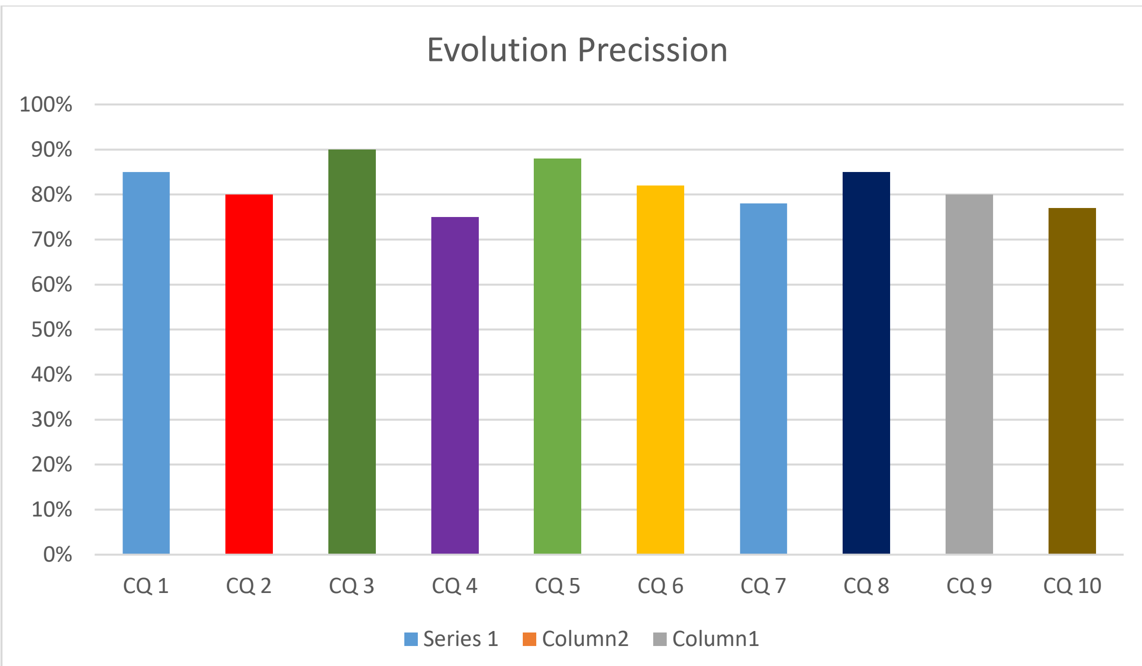

Figure 4: Success Rate Comparison Chat.

Table 1 shows possible Competency Questions (CQ's) that can trigger concepts that are not manually encoded in the existing ontology, Candidate Terms in the Search Query String used in filtering and collecting related documents from the secondary data source (Wikipedia). The Multi-Concept Ontology Evolution framework generated newly multiple concept not on the Ontology Knowledge based on CQ's, existing MaCOnto file, and Wikipedia. Concept generated includes '*FertilizerEffectOnYield*', '*IrrigationEffectOnGrowth*', '*MaizeDiseaseTypes*', and the likes. lastly, the performance of the Multi-concept is calculated and evaluated based on the expected concept and the generated concept using Equation 1.

$$Sucess\ Rate = \frac{Number\ of\ Successful\ Outcomes}{Total\ Number\ of\ Outcomes} \times 100 \text{……………….Eq(1)}$$

Averagely the Multi-Concept based ontology Evolution framework yielded an average performance of 83.5% success rate.

References

Al-Aswadi, F. N., Chan, H. Y., & Gan, K. H. (2020). Automatic ontology construction from text: A review from shallow to deep learning trend. *Artificial Intelligence Review*, *53*(6), 3901–3928. https://doi.org/10.1007/s10462-019-09782-9

Aminu, E. F., Oyefolahan, I. O., Abdullahi, M. B., & Salaudeen, M. T. (2022). MaCOnto: A robust maize crop ontology based on soils, fertilizers and irrigation knowledge. *Intelligent Systems with Applications*, *16*, 200125.

Babaei Giglou, H., D'Souza, J., & Auer, S. (2023). LLMs4OL: Large Language Models for Ontology Learning. In T. R. Payne, V. Presutti, G. Qi, M. Poveda-Villalón, G. Stoilos, L. Hollink, Z. Kaoudi, G. Cheng, & J. Li (Eds.), *The Semantic Web – ISWC 2023* (Vol. 14265, pp. 408–427). Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-47240-4_22

Bunnell, L., Osei-Bryson, K.-M., & Yoon, V. Y. (2021). Development of a consumer financial goals ontology for use with FinTech applications for improving financial capability. *Expert Systems with Applications*, *165*, 113843.

Cardoso, S. D., Da Silveira, M., & Pruski, C. (2020). Construction and exploitation of an historical knowledge graph to deal with the evolution of ontologies. *Knowledge-Based Systems*, *194*, 105508. https://doi.org/10.1016/j.knosys.2020.105508

Dou, D., Wang, H., & Liu, H. (2015). Semantic data mining: A survey of ontology-based approaches. *Proceedings of the 2015 IEEE 9th International Conference on Semantic Computing (IEEE ICSC 2015)*, 244–251. https://ieeexplore.ieee.org/abstract/document/7050814/

Gruber, T. (1993). *What is an Ontology*. https://queksiewkhoon.tripod.com/ontology_01.pdf

Hassan, S.-U., Saleem, A., Soroya, S. H., Safder, I., Iqbal, S., Jamil, S., Bukhari, F., Aljohani, N. R., & Nawaz, R. (2021). Sentiment analysis of tweets through Altmetrics: A machine learning approach. *Journal of Information Science*, *47*(6), 712–726. https://doi.org/10.1177/0165551520930917

Hawley, S. H. (2019). *Challenges for an Ontology of Artificial Intelligence* (No. arXiv:1903.03171). arXiv. http://arxiv.org/abs/1903.03171

Khadir, A. C., Aliane, H., & Guessoum, A. (2021). Ontology learning: Grand tour and challenges. *Computer Science Review*, *39*, 100339. https://doi.org/10.1016/j.cosrev.2020.100339

Kotis, K. I., Vouros, G. A., & Spiliotopoulos, D. (2020). Ontology engineering methodologies for the evolution of living and reused ontologies: Status, trends, findings and recommendations. *The Knowledge Engineering Review*, *35*, e4.

Larentis, A. V., Neto, E. G. de A., Barbosa, J. L. V., Barbosa, D. N. F., Leithardt, V. R. Q., & Correia, S. D. (2021). Ontology-based reasoning for educational assistance in noncommunicable chronic diseases. *Computers*, *10*(10), 128.

Lee, Y.-C., Eastman, C. M., & Solihin, W. (2016). An ontology-based approach for developing data exchange requirements and model views of building information modeling. *Advanced Engineering Informatics*, *30*(3), 354–367.

Navarro-Almanza, R., Juárez-Ramírez, R., Licea, G., & Castro, J. R. (2020). Automated Ontology Extraction from Unstructured Texts using Deep Learning. In O. Castillo, P. Melin, & J. Kacprzyk (Eds.), *Intuitionistic and Type-2 Fuzzy Logic Enhancements in Neural and Optimization Algorithms: Theory and Applications* (pp. 727–755). Springer International Publishing. https://doi.org/10.1007/978-3-030-35445-9_50

Osman, M. A., Noah, S. A. M., & Saad, S. (2022). Ontology-based knowledge management tools for knowledge sharing in organization—A review. *IEEE Access*, *10*, 43267–43283.

Pietranik, M., & Kozierkiewicz, A. (2023). Methods of managing the evolution of ontologies and their alignments. *Applied Intelligence*, *53*(17), 20382–20401. https://doi.org/10.1007/s10489-023-04545-0

Safyan, M., Ul Qayyum, Z., Sarwar, S., Iqbal, M., Garcia Castro, R., & Al-Dulaimi, A. (2019). Ontology evolution for personalised and adaptive activity recognition. *IET Wireless Sensor Systems*, *9*(4), 193–200. https://doi.org/10.1049/iet-wss.2018.5209

Smith, B. (2012). Ontology. In *The furniture of the world* (pp. 47–68). Brill. https://brill.com/downloadpdf/display/book/edcoll/9789401207799/B9789401207799-s005.pdf

Tudorache, T. (2020). Ontology engineering: Current state, challenges, and future directions. *Semantic Web*, *11*(1), 125–138.

Urdaneta-Ponte, M. C., Méndez-Zorrilla, A., & Oleagordia-Ruiz, I. (2021). Lifelong learning courses recommendation system to improve professional skills using ontology and machine learning. *Applied Sciences*, *11*(9), 3839.

Yang, L., Li, Y., Wang, J., & Sherratt, R. S. (2020). Sentiment Analysis for E-Commerce Product Reviews in Chinese Based on Sentiment Lexicon and Deep Learning. *IEEE Access*, *8*, 23522–23530. https://doi.org/10.1109/ACCESS.2020.2969854