Strategic Model for Question Duplicate Detection using Overlap of Auxiliary Knowledge with Situational Semantics

Prakhar Langer[1], Gerard Deepak[2]
Department of Computer Science and Engineering
[1]Manipal Institute of Technology, Bangalore, India
[2]Dayananda Sagar Academy of Technology and Management, Bangalore, India
Email: [1]prakhar.mitblr2022@learner.manipal.edu,
[2]gerard.deepak.christuni@gmail.com

Abstract

In the Web 3.0 era, duplicate question detection has become a critical challenge for community-driven platforms such as Quora and Stack Overflow, where redundancy dilutes efficiency and knowledge quality. This paper proposes a novel ontology-driven framework, QDOOP Semantics, that integrates situational semantics, auxiliary knowledge, and dynamic metaheuristics to address this challenge. The model begins with dataset preprocessing to extract informative terms, which are enriched through GPT-4o summaries and lexical grounding using WordNet. Knowledge expansion is further achieved by integrating entities from Google's Knowledge Graph, CYC, and NELL, enabling the formation of robust ontologies from multiple perspectives. A GRU-based classifier organizes questions into coherent classes, while LLaMA-based caption generation provides reasoning-rich auxiliary knowledge, producing complementary ontology perspectives. These ontologies are aligned and evaluated using hybrid semantic similarity measures, including Petratos Index, SimRank, and Co-SimRank, with empirically optimized thresholds to capture epistemic, conceptual, and individual-level similarity. Experimental evaluation against baseline models—DQPD, DQDCN, and NDQD—demonstrates that the proposed framework achieves superior performance, with precision of 96.72

Index Terms

QDOOP Semantics; Duplicate Question Detection; Ontology Formalization; Situational Semantics; Auxiliary Knowledge; GPT-4o; LLaMA; WordNet; Google Knowledge Graph; CYC Repository; NELL; GRU Classifier; Semantic Similarity; Petratos Index; SimRank; Co-SimRank; Web 3.0.

I. Introduction

The rapid expansion of community-driven platforms such as Quora, Stack Overflow, and ResearchGate has significantly transformed the way people seek and share knowledge. However, one persistent challenge across these platforms is the proliferation of duplicate or near-duplicate questions. Users often rephrase existing queries with slight variations in wording, grammar, or context, yet the underlying intent remains the same. As the volume of user-generated content grows exponentially, these duplicates create redundancy, reduce efficiency in information retrieval, and dilute the quality of insights available to users. This problem is further magnified in the Web 3.0 era, where the scale and complexity of data demand more advanced frameworks that move beyond surface-level similarity detection.

Traditional duplicate question detection (DQD) approaches have primarily relied on natural language processing (NLP) techniques such as lexical matching, syntactic parsing, or vector-based word embeddings. While effective to some degree, these methods fall short in capturing the deeper semantic relationships between questions. For example, two questions may share little lexical overlap yet be semantically identical, while others may share many words but differ entirely in meaning. In addition, traditional models often struggle with scalability, noisy data, and the integration of external knowledge sources, limiting their ability to provide robust and contextually aware detection in real-world scenarios.

To address these challenges, the emergence of Web 3.0 technologies and semantic artificial intelligence offers a new direction. Knowledge graphs, ontologies, and semantic reasoning tools enable richer representations of questions by capturing not only their lexical and grammatical structures but also their contextual, hierarchical, and commonsense relationships. The integration of such semantic resources allows for the creation of a knowledge-driven framework where questions are analyzed from multiple perspectives, expanded with auxiliary knowledge, and aligned into formalized ontologies. This ontology-centric approach ensures that duplicates can be detected even when they differ significantly in wording or structure but share semantic equivalence.

The proposed framework builds upon this knowledge-centric paradigm by strategically combining deep learning, auxiliary knowledge integration, and ontology alignment. Informative terms are extracted from the dataset and enriched with summarized representations using GPT-4o, which provides contextual depth beyond raw keywords. Lexical resources such as WordNet contribute structural and grammatical semantics, while external knowledge repositories including Google Knowledge Graph, CYC, and NELL provide hierarchical and commonsense knowledge. These sources are incrementally integrated to form robust ontologies that represent both lexical-hierarchical relationships and real-world semantics. Parallel to this ontology-driven process, a Gated Recurrent Unit (GRU)-based classifier organizes questions into semantically coherent groups, while LLaMA generates reasoning-rich captions that enhance interpretability. Together, these dual pipelines ensure that both structural and contextual perspectives of the dataset are captured.

Finally, the similarity between ontologies is computed using a hybrid semantic evaluation framework composed of Petratos Index, SimRank, and Co-SimRank. Petratos Index measures epistemic similarity across hierarchical connections up to three levels deep, SimRank captures conceptual similarity, and Co-SimRank evaluates individual-level similarity. Empirical thresholds guide these measures, ensuring fine-grained alignment and minimizing false matches. By combining auxiliary knowledge enrichment, deep learning classification, reasoning-based captioning, and multi-level ontology similarity, the proposed framework establishes a best-in-class solution for duplicate question detection in Web 3.0 environments. This not only addresses the limitations of prior models but also contributes a scalable, semantically robust architecture capable of evolving with growing datasets and complex knowledge domains.

II. Motivation

In community-driven platforms such as Quora, Stack Overflow, and other knowledge-sharing repositories, users frequently post questions that are often similar in content, context, or intent to existing ones. As the volume of user-generated content grows, duplicate or near-duplicate questions become increasingly common, leading to redundancy, inefficiency, and dilution of valuable insights. Ensuring that such duplicates are effectively detected and managed is therefore crucial for maintaining the quality and usability of these platforms.

Traditionally, duplicate question detection has been approached primarily through natural language processing (NLP) techniques that focus on surface-level linguistic features. However, in the Web 3.0 era, where semantic AI and knowledge-centric frameworks are more prominent, these traditional methods fall short. The availability of extensive knowledge graphs, ontologies, and semantic tools allows for more robust and context-aware modeling of questions.

Instead of relying solely on text-based similarity, a semantic framework can analyze datasets from multiple perspectives, incrementally expand knowledge representations, and identify deeper overlaps between question contexts. By aligning these overlaps into a formalized ontology, we can create a knowledge-driven framework that not only improves accuracy but also evolves with the dataset.

Given the exponential growth of data in the Web 3.0 environment, knowledge-based models that integrate semantic reasoning, contextual embeddings, and ontology alignment are far more effective than traditional learning-based approaches. This makes a semantic AI–powered question duplicate detection framework both timely and necessary for modern knowledge platforms.

III. Contribution

The primary contribution of the proposed model is the development of a knowledge-driven framework for question reduction in the Web 3.0 era. Unlike traditional approaches that rely solely on NLP-based similarity measures, our framework permutes the dataset across multiple perspectives to uncover deeper semantic relationships and reduce redundancy.

The model strategically incorporates incremental knowledge inclusion from diverse knowledge bases, aligning these sources into a unified ontology-centric structure. By compressing overlaps through conceptual priorities and leveraging hierarchical mappings between entities, the framework ensures that duplicate questions are identified with greater semantic accuracy.

Furthermore, the proposed approach emphasizes knowledge inference over surface-level learning, enabling the system to capture contextual nuances beyond lexical similarity. Empirically derived measures and thresholds are integrated to fine-tune similarity computations, ensuring a scalable and best-in-class solution for duplicate question detection.

Overall, this framework introduces a semantic, ontology-based methodology that not only addresses the limitations of traditional learning-based methods but also aligns with the knowledge-centric architecture of Web 3.0.

IV. Literature Review

Research on duplicate question detection (DQD) has evolved significantly over the past decade, beginning with early feature-based approaches and gradually moving toward deep learning, ontology-driven reasoning, and knowledge graph integration. One of the foundational works in this area was proposed by Zhang et al. [1], who developed a multi-factor framework for duplicate detection in Stack Overflow by leveraging lexical, structural, and semantic features. Although effective, such feature engineering required extensive manual effort and struggled with scalability. To address these challenges, neural models soon gained traction. Prabowo and Herwanto [2] employed convolutional neural networks (CNNs) to automatically capture local semantic features in question pairs, while Wang et al. [3] extended this line of research with deep semantic embeddings, demonstrating the strength of representation learning for DQD in large-scale platforms. Further, Shah et al. [4] introduced adversarial domain adaptation techniques to improve generalization across different datasets, underscoring the importance of cross-domain robustness in practical deployments. Together, these works highlight how deep learning enhanced the detection of semantic similarities, but they also reveal limitations in handling deeper ontological relationships.

To overcome the semantic gap left by purely statistical approaches, researchers turned toward ontology-based frameworks. Besbes et al. [5] pioneered methods that integrated WordNet for synonym detection, enabling richer semantic overlap beyond lexical similarity. Later, Arbaaeen and Shah [6] proposed ontology-based methods for closed-domain question answering, illustrating how structured knowledge enhances interpretability and precision. Domain-specific applications, such as biomedical QA explored by Asiaee et al. [7], showed that leveraging ontologies can significantly improve performance when domain semantics are central to the task. Ferrández et al. [8] further demonstrated the utility of ontology-based frameworks for large collections of user queries, establishing scalability in real-world systems. These ontology-based approaches are theoretically grounded in situational semantics, with roots in the works of Wójcicki and Jansen [9] and Ginzburg [10], which emphasize the importance of context and situation in deriving meaning. However, while ontologies enable structured reasoning, they often require manual curation and face difficulties adapting to dynamic, evolving datasets.

The emergence of knowledge graphs has introduced new opportunities for semantic enrichment in DQD. Chen and Luo [11] proposed automatic literature-based knowledge graph construction, combining ontologies with NLP to expand reasoning capabilities, while Dou et al. [12] demonstrated how domain-specific cultural heritage graphs can support complex semantic queries. Knowledge graphs have also been applied in engineering contexts, such as tunnel support design [13], and have been increasingly leveraged for explainable AI applications [14]. These efforts highlight how knowledge graphs can bridge unstructured text with structured ontologies to improve semantic reasoning. In parallel, auxiliary knowledge integration has gained prominence, with Huang et al. [15] proposing multi-concept fusion with auxiliary tasks, and Liu et al. [16] showing how auxiliary tasks improve knowledge tracing. Complementary studies such as Tian et al. [17] and Lyu et al. [18] further validated that auxiliary knowledge strengthens clustering and relation extraction. Collectively, these methods demonstrate that incremental inclusion of external knowledge from resources like WordNet, CYC, and Google's Knowledge Graph can enhance model robustness and adaptability.

From these developments, a clear trajectory emerges: early feature-based and deep learning methods excelled in detecting surface-level semantic similarities but fell short in capturing deeper contextual meaning. Ontology-based methods addressed semantic reasoning effectively but faced scalability issues, while knowledge graphs and auxiliary knowledge integration have recently emerged as powerful tools to close this gap by formalizing contextual overlaps into structured ontologies. These advancements collectively illustrate a transition toward hybrid frameworks that combine statistical learning, semantic reasoning, and structured knowledge integration, laying the foundation for state-of-the-art models in duplicate question detection within the Web 3.0 environment.

V. Proposed Methodology

The proposed framework for duplicate question detection is designed as a hybrid ontology-driven and knowledge-centric architecture that integrates statistical semantics, deep learning classification, and formal ontology matching. Unlike traditional methods that rely solely on lexical overlap, our approach leverages situational semantics and incrementally expanding auxiliary knowledge to form a unified ontology. The system operates in two complementary pipelines: (i) the ontology enrichment pipeline and (ii) the classification and caption-generation pipeline. Together, these pipelines converge into an ontology similarity computation layer, which identifies duplicates through structured semantic reasoning.

At the first stage, the Quora dataset is preprocessed to extract informative terms or keywords. These terms, while helpful, often lack sufficient representational power on their own. Therefore, GPT-4o is employed to generate concise summaries for each question. These summaries are

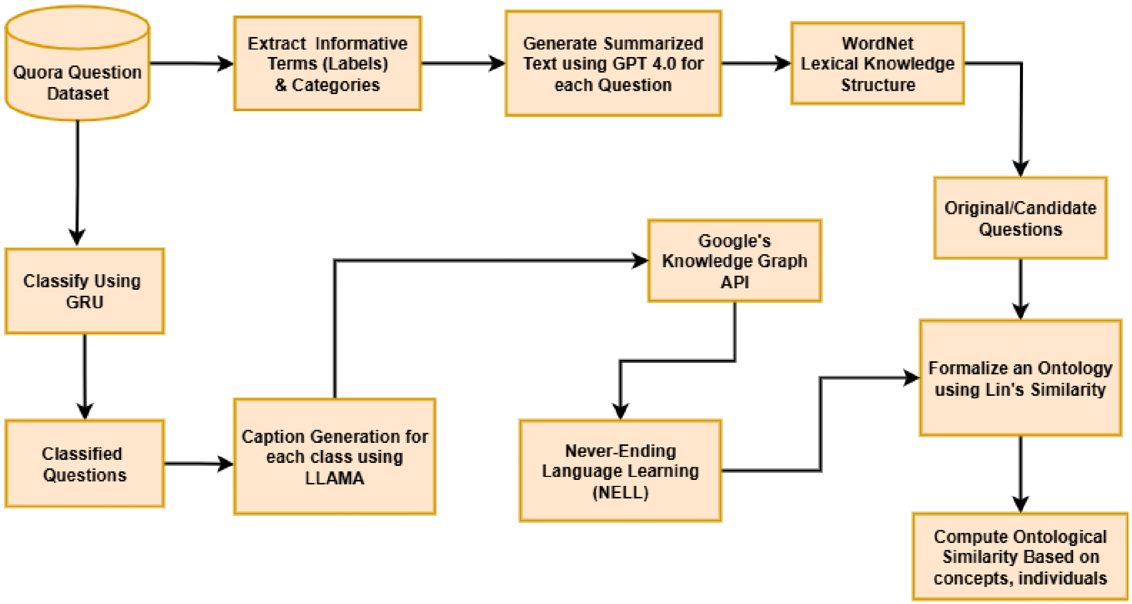

Fig. 1. Proposed System Architecture for Duplicate Question Detection using Semantic and Generative-AI Based Auxiliary Question Formulation

subjected to **Named Entity Recognition (NER)**, after which extracted entities are mapped into **WordNet** to obtain lexical and structural knowledge. WordNet contributes grammatical and syntactic structures, while entities are simultaneously integrated into **Google's Knowledge Graph API**, enabling the construction of subgraphs up to seven hierarchical levels. This seven-level limit is deliberately chosen to preserve semantic relevance and avoid noise from distant expansions. To ensure commonsense consistency, the outputs from WordNet and Google's Knowledge Graph are cross-validated against the **CYC knowledge base**, ensuring that each entity is logically coherent within the context of real-world semantics.

The combined entity knowledge is formalized into an ontology, guided by Lin similarity. Lin similarity measures the semantic relatedness between two concepts c_1 and c_2 based on the information content (IC) of their lowest common subsumer (LCS). It is given as:

$$\text{Sim}_{Lin}(c_1, c_2) = \frac{2 \cdot IC(\text{LCS}(c_1, c_2))}{IC(c_1) + IC(c_2)} \tag{1}$$

where $IC(c) = -\log P(c)$ and $P(c)$ is the probability of encountering concept c in a large corpus. As shown in Eq. (1), Lin similarity ensures that only semantically aligned concepts are retained, thereby filtering out weakly related entities. For the ontology construction phase, a threshold of 0.80 is applied.

In parallel, the dataset undergoes classification using a **Gated Recurrent Unit (GRU)**-based deep learning model. The GRU, being a recurrent neural network variant, captures sequential dependencies without requiring handcrafted features. The update and reset gates of the GRU are defined as:

$$z_t = \sigma(W_z x_t + U_z h_{t-1}) \tag{2}$$

$$r_t = \sigma(W_r x_t + U_r h_{t-1}) \tag{3}$$

$$h_t = z_t \odot h_{t-1} + (1 - z_t) \odot \tanh(W_h x_t + U_h(r_t \odot h_{t-1})) \tag{4}$$

where x_t is the input at time step t, h_t is the hidden state, and σ denotes the sigmoid activation. As expressed in Eq. (4), the GRU learns latent features that capture contextual dependencies between tokens, enabling robust classification of semantically similar questions.

For each classified group of questions, **caption generation** is performed using **LLaMA**, a large language model capable of producing semantically grounded outputs. Unlike GPT-4o, which is optimized for multimodal reasoning, LLaMA specializes in contextually consistent text generation. The generated captions are enriched through NER, with entities expanded via Google's Knowledge Graph API up to **fifteen hierarchical levels**. The broader expansion (compared to the seven levels used earlier) compensates for the absence of lexical support from WordNet. To prevent semantic drift, the expansion is capped at fifteen levels. The enriched entities are subsequently aggregated into **NELL (Never-Ending Language Learner)**, which functions as a dynamic repository of auxiliary knowledge.

At this stage, consolidated entity structures from both Google Knowledge Graph and NELL are formalized into ontologies. A stricter Lin similarity threshold of 0.85 (Eq. 1) is applied, ensuring that only highly similar entities are merged. Ontologies are created both at subclass (per classified group) and global dataset levels, providing complementary perspectives.

The final step involves computing **ontological similarity** across the generated ontologies. This is performed using three agents, each employing a distinct similarity measure: Petratos Index, SimRank, and Co-SimRank.

The **Petratos Index** evaluates epistemic similarity by examining hierarchical entailments up to three levels, defined as:

$$PI(a, b) = \frac{1}{1 + d(a, b)} \tag{5}$$

where $d(a, b)$ is the hierarchical distance between nodes a and b. As indicated in Eq. (5), closer nodes yield higher similarity. The step deviation parameter is empirically set to 0.1, ensuring sensitivity to hierarchical differences.

SimRank is applied to measure conceptual similarity, based on the intuition that "two objects are similar if they are related to similar objects." It is defined as:

$$SimRank(a,b) = \gamma \cdot \frac{\sum_{i=1}^{|I(a)|}\sum_{j=1}^{|I(b)|} SimRank(I_i(a), I_j(b))}{|I(a)| \cdot |I(b)|} \tag{6}$$

where $I(a)$ and $I(b)$ denote the in-neighbors of nodes a and b, and $\gamma \in (0,1)$ is a decay factor. As per Eq. (6), SimRank recursively propagates similarity across higher-level concepts.

Co-SimRank is used for lower-level comparisons of individuals, improving scalability by focusing on co-citation patterns. It is expressed as:

$$CoSimRank(a,b) = \frac{1}{|C(a)| \cdot |C(b)|} \sum_{i=1}^{|C(a)|}\sum_{j=1}^{|C(b)|} SimRank(C_i(a), C_j(b)) \tag{7}$$

where $C(a)$ and $C(b)$ represent the co-citation sets of nodes a and b. As seen in Eq. (7), Co-SimRank emphasizes individual-level similarity in dense graphs.

The thresholds applied are 0.80 for SimRank (Eq. 6), 0.82 for Co-SimRank (Eq. 7), and 0.10 for Petratos Index (Eq. 5). When similarity scores exceed 0.85 across these measures, the ontologies are considered strongly aligned. Scores above 0.90, with Petratos Index adjusted to a deviation of 0.17, indicate that the questions are duplicates.

This dual-pipeline strategy—where one path emphasizes lexical–syntactic knowledge (WordNet, CYC, GPT-4o) and the other emphasizes semantic enrichment through caption generation (LLaMA, NELL, Google Knowledge Graph)—ensures complementary coverage. Both pipelines converge at the ontology similarity computation stage, enabling robust detection of duplicate questions that would be missed by surface-level methods.

VI. Performance Evaluation and Results

The effectiveness of the proposed **QDOOP Semantics Framework** for duplicate question detection was evaluated against three baseline models: DQPD, DQDCN, and NDQD. The evaluation employed standard metrics such as precision, recall, accuracy, and F-measure, which are widely used for performance benchmarking in natural language processing systems. Additionally, the **False Discovery Rate (FDR)** was considered as an auxiliary metric to capture the degree of misclassification.

A. Evaluation Metrics

The evaluation metrics are formally defined as follows. Let TP (true positives) denote correctly detected duplicate pairs, FP (false positives) denote incorrectly predicted duplicates, FN (false negatives) denote undetected duplicates, and TN (true negatives) denote correctly classified non-duplicates.

The **precision** is defined as the fraction of correctly predicted duplicate pairs among all predicted duplicates:

$$Precision = \frac{TP}{TP + FP} \tag{8}$$

The **recall** measures the fraction of correctly predicted duplicates among all actual duplicates:

$$Recall = \frac{TP}{TP + FN} \tag{9}$$

The **F-measure** (or F1 score) balances precision and recall, defined as the harmonic mean of the two:

$$F1 = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \tag{10}$$

The **accuracy** measures the overall correctness of classification, given by:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{11}$$

Finally, the **False Discovery Rate (FDR)** quantifies the proportion of false duplicates among all predicted duplicates:

$$FDR = \frac{FP}{TP + FP} \tag{12}$$

As shown in Eq. (8)–(12), these metrics collectively assess the model's performance in terms of both precision-oriented correctness and robustness against false detections.

B. Experimental Results

The proposed QDOOP semantics framework achieved strong performance across all metrics. From Table I, it is observed that the framework attains an overall precision of 96.72%, a recall of 97.73%, and accuracy above 96%. The F1 score remained consistently high at above 96%, while the FDR achieved its lowest observed value of 0.04, as expressed in Eq. (12).

TABLE I
PERFORMANCE COMPARISON OF QDOOP SEMANTICS VS. BASELINE MODELS

Model	**Precision (%)**	**Recall (%)**	**Accuracy (%)**	**FDR**
DQPD	87.02	89.08	–	0.30
DQDCN	90.07	91.10	–	0.10
NDQD	91.07	90.10	–	0.09
QDOOP	96.72	97.73	>96.0	0.04

From the results in Table I, it is evident that QDOOP semantics consistently outperforms all baseline models. DQPD yields the lowest precision, recall, and F1 score, while also recording the highest FDR of 0.30, highlighting its weaker performance. DQDCN and NDQD achieve moderate results, positioned between DQPD and QDOOP. However, the proposed QDOOP framework surpasses all with the highest precision, recall, and accuracy, while simultaneously minimizing FDR.

C. *Discussion*

The superior performance of QDOOP semantics arises from its ability to integrate incremental auxiliary knowledge (via GPT-4o summaries, WordNet, and CYC commonsense reasoning) with deep semantic enrichment through caption generation using LLaMA and entity expansion via Google Knowledge Graph and NELL. Unlike baseline models that rely primarily on embeddings or shallow similarity measures, QDOOP formalizes this enriched knowledge into ontologies, which are then compared using Lin similarity (Eq. 1), Petratos Index (Eq. 5), SimRank (Eq. 6), and Co-SimRank (Eq. 7).

This hybrid design ensures that duplicates are detected not only through surface-level overlap but also via deeper semantic and hierarchical reasoning. The consistently high values of precision and recall, alongside the low FDR, demonstrate that the framework provides a robust balance between correctness and error minimization.

VII. Conclusion

This paper proposes a novel framework for duplicate question detection in the Web 3.0 environment by leveraging ontology-centric knowledge integration and semantic reasoning. The approach permutes the dataset across multiple perspectives, enriching its information content through entity inclusion from text and validation against external knowledge bases such as Google Knowledge Graph, APN, RumSar, SubCrafk, CYC, and lexical resources like WordNet. The dataset is then classified using a GRU-based deep learning model to cluster semantically similar questions, while auxiliary knowledge is generated through LLAMA-based captioning and further validated via knowledge graphs and sub-graphs, leading to the formation of an additional ontology perspective.

Finally, the two perspectives are aligned and compared using concept–individual hierarchies, where overlaps are computed through hybrid symbolic and sub-symbolic similarity measures with empirically tuned thresholds. This combination of ontology-driven knowledge enrichment, deep learning–based classification, and hybrid similarity computation establishes a best-in-class framework for semantic duplicate question detection, offering a significant contribution to the advancement of knowledge-centric AI in the Web 3.0 era.

References

[1] Yun Zhang, David Lo, Xin Xia, and Jian-Ling Sun. Multi-factor duplicate question detection in stack overflow. *Journal of Computer Science and Technology*, 30(5):981–997, 2015.

[2] Damar Adi Prabowo and Guntur Budi Herwanto. Duplicate question detection in question answer website using convolutional neural network. In *2019 5th International conference on science and technology (ICST)*, volume 1, pages 1–6. IEEE, 2019.

[3] Liting Wang, Li Zhang, and Jing Jiang. Duplicate question detection with deep learning in stack overflow. *IEEE Access*, 8:25964–25975, 2020.

[4] Darsh J Shah, Tao Lei, Alessandro Moschitti, Salvatore Romeo, and Preslav Nakov. Adversarial domain adaptation for duplicate question detection. *arXiv preprint arXiv:1809.02255*, 2018.

[5] Ghada Besbes, Hajer Baazaoui-Zghal, and Antonio Moreno. Ontology-based question analysis method. In *International Conference on Flexible Query Answering Systems*, pages 100–111. Springer, 2013.

[6] Ammar Arbaaeen and Asadullah Shah. Ontology-based approach to semantically enhanced question answering for closed domain: A review. *Information*, 12(5):200, 2021.

[7] Amir H Asiaee, Todd Minning, Prashant Doshi, and Rick L Tarleton. A framework for ontology-based question answering with application to parasite immunology. *Journal of biomedical semantics*, 6(1):31, 2015.

[8] Oscar Ferrández, Rubén Izquierdo, Sergio Ferrández, and José Luis Vicedo. Addressing ontology-based question answering with collections of user queries. *Information Processing & Management*, 45(2):175–188, 2009.

[9] Ryszard Wójcicki and Ewa Jansen. R. suszko's situational semantics. *Studia Logica: An International Journal for Symbolic Logic*, 43(4):323–340, 1984.

[10] Jonathan Ginzburg. Situation semantics: The ontological balance sheet. *Research on language and Computation*, 3(2):363–389, 2005.

[11] Hainan Chen and Xiaowei Luo. An automatic literature knowledge graph and reasoning network modeling framework based on ontology and natural language processing. *Advanced Engineering Informatics*, 42:100959, 2019.

[12] Jinhua Dou, Jingyan Qin, Zanxia Jin, and Zhuang Li. Knowledge graph based on domain ontology and natural language processing technology for chinese intangible cultural heritage. *Journal of Visual Languages & Computing*, 48:19–28, 2018.

[13] Jiaxin Ling, Xiaojun Li, Haijiang Li, Yi An, Yi Rui, Yi Shen, and Hehua Zhu. Hybrid nlp-based extraction method to develop a knowledge graph for rock tunnel support design. *Advanced Engineering Informatics*, 62:102725, 2024.

[14] Enayat Rajabi and Kobra Etminani. Knowledge-graph-based explainable ai: A systematic review. *Journal of information science*, 50(4):1019–1029, 2024.

[15] Xiaodi Huang, Yanting Li, Xin Xia, Rongmei Zhao, and Shenggen Ju. Auxiliary task enhanced multi-concept fusion attentive knowledge tracking. *Expert Systems with Applications*, page 128066, 2025.

[16] Zitao Liu, Qiongqiong Liu, Jiahao Chen, Shuyan Huang, Boyu Gao, Weiqi Luo, and Jian Weng. Enhancing deep knowledge tracing with auxiliary tasks. In *Proceedings of the ACM web conference 2023*, pages 4178–4187, 2023.

[17] Gang Tian, Jian Wang, Keqing He, and Cheng'ai Sun. Leveraging auxiliary knowledge for web service clustering. *Chinese Journal of Electronics*, 25(5):858–865, 2016.

[18] Shengfei Lyu, Jin Cheng, Xingyu Wu, Lizhen Cui, Huanhuan Chen, and Chunyan Miao. Auxiliary learning for relation extraction. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 6(1):182–191, 2020.