International Journal on Advanced Science Engineering Information Technology

Vol.14 (2024) No. 4
ISSN: 2088-5334

Application of Machine Learning in the Web of Linked Data

Yuri Kim [a], Jihee Lee [a], Seoyeon Oh [a], Jihyun Kim [b], Jeongin Mok [a], Chaerin Noh [a] , Seongbin Park [a,*]

[a] Department of Computer Science, Korea University, Sungbook-ku, An-am-ro 145, Seoul, Republic of Korea
[b] Department of Computer Science, University of Pennsylvania,3451 Walnut Street, Philadelphia, United States
Corresponding author: [*]hyperspace@korea.ac.kr

Abstract— **Linked Open Data (LOD) refers to guidelines for publishing and connecting structured data on the internet. Utilizing web technologies like HTTP, RDF, and URIs, Linked Data establishes entities across diverse domains and links them through categorized connections, thus forming a web of data readable by machines rather than humans. The LOD, often dubbed the Web of Linked Data, is an ever-expanding realm of information. Beyond mere data accumulation, the LOD methodology involves establishing connections between datasets. LODs and ontologies offer a universal solution that facilitates system interoperability, allowing for the sharing and utilizing shared information. However, since not all LODs employ the same ontologies, the use of diverse vocabularies and ontologies by organizations and communities across different fields to formalize entities and their relationships poses challenges to interoperability between different sets of LODs. When integrating LODs from various ontologies into a single entity, missing links may arise, leading to what we refer to as missing link scenarios. This paper examines multiple missing link scenarios primarily stemming from scattered ontologies across LODs. Subsequently, we propose feature- and graph-based methods for identifying missing links between LODs, significantly leveraging diverse ontologies. This research aims to provide a comprehensive review and introduce missing link management in LODs, which can facilitate the discovery of more valuable data by establishing connections with other datasets and enabling its more effective utilization through inference and semantic queries and rules.**

Keywords— **Linked data; hyperlink; machine learning.**

Manuscript received 15 Dec. 2023; revised 29 Apr. 2024; accepted 2 Aug. 2024. Date of publication 31 Aug. 2024.

I. INTRODUCTION

Throughout the annals of technology, the Internet stands out as one of the most transformative breakthroughs in human communication [1]. Over the past two decades, there has been an exponential surge in the volume of data. Traditional data management systems cannot store and process this vast web data. Linked Open Data (LOD) emerges as an extension of the current World Wide Web (WWW), offering a standardized approach to recycle, exchange, and publish data across applications and communities (e.g., Fig. 1) [2].

By leveraging web technologies like HTTP, RDF, and URIs, Linked Data establishes connections between entities spanning diverse domains, creating a machine-readable web of data [3]. LODs rely on ontologies to structure and organize data within their framework. Ontologies serve as machine-readable depictions of knowledge within a specific application domain, typically delineated in a declarative knowledge modeling language like OWL (Web Ontology Language) [4], which relies on description logic (DL).

Within ontologies, entities include individuals, classes (groupings of individuals), and properties (relationships between individuals), with semantics defined through a series of logical statements known as axioms. Moreover, ontology effectively defines various data objects, potentially finding widespread application in big data environments. LODs and ontologies offer a universal solution that enables systems to interoperate, facilitating the sharing and utilization of information. This diminishes or eliminates the need for manual information exchange, as system interactions can be automated [5].

LODs foster an ecosystem of interconnected data and information, allowing data resources to be explicitly or implicitly linked to others [6]. This is possible through shared naming and equivalence statements across web repositories [7]. LODs enable discovering additional valuable data by establishing connections with other datasets. Moreover, they empower the more effective exploitation of this data through inferencing, semantic queries, and rules [8].

The potential of LODs is vast: search engines can flourish, definitions can be enhanced, and exploratory searches can be facilitated. The connections between datasets enable references to similar entities and the reuse of their descriptions, consolidating scattered facts and providing a broader perspective or accessing additional information beyond what a single dataset could offer. With advancements in Artificial Intelligence, leveraging such data inputs has become feasible, aiding in the generation, curation, sharing, and maintenance of corpora and datasets [9].

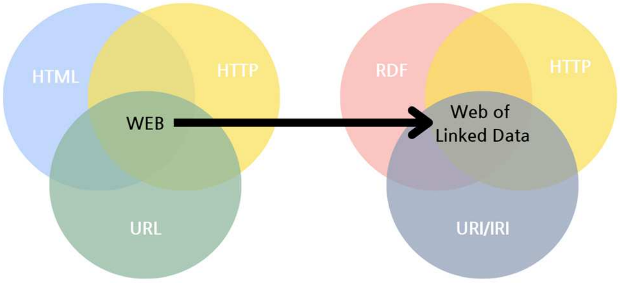


Fig. 1 Transition from the Web to Web of Linked Data

The LOD cloud encompasses over 1000 available datasets, some containing billions of triples. As of 2024, the cloud comprises 1314 datasets interconnected by 16308 links, spanning domains such as Geography, Government, Life Sciences, Linguistics, Media, Publications, Social Networking, and User Generated, as shown in Fig. 2.

There are four prominent cross-domain datasets within the LOD ecosystem: DBpedia, Wikidata, Freebase, and YAGO. DBpedia serves as a knowledge base to make the wealth of information available on Wikipedia (whether unstructured or semi-structured) [10]. The user community actively contributes by establishing mappings that link information representations in Wikipedia with the DBpedia ontology. As one of the foundational nodes of the LOD cloud, DBpedia has been consistently maintained since its inception. It encompasses information in 125 languages, with the English segment alone containing over 28 million triples across various domains. Wikidata aims to offer a machine-readable representation of knowledge within Wikimedia projects [11].

This project consolidates all languages from Wikimedia projects into a unified, easily accessible interface. Wikidata serves as a community-maintained knowledge base where users with accounts can contribute by adding, updating, or deleting triples. Currently, it contains over 93 million interlinked entities. Freebase shares a similar vision with DBpedia regarding knowledge extraction from Wikipedia but adopts a broader perspective by incorporating data from other sources [12]. Despite its similarity to Wikidata regarding user-driven updates, Freebase was managed by domain experts rather than community members. YAGO positions itself as a refined, streamlined version of Wikidata, striving to enhance its usability and reliability by enforcing a strict type hierarchy with semantic constraints [13]. Initially combining data from Wikipedia and Wordnet, YAGO is transitioning to its fourth version, which integrates Wikidata and schema.org.

Various LOD collections maintained by different organizations and communities exhibit a diversity of ontologies. A significant challenge within the LOD context arises from the fragmented nature of ontologies. More specifically, one such challenge involves the overlap and ambiguity of concepts, exacerbated by inconsistent usage of properties like "subclass of" and "instance of" [13].

For example, the term "Scientist" can represent both a profession and an individual practicing science, leading to the entity "Scientist" being classified as both a subclass of "person" and an instance of "profession." Additionally, discrepancies arise in defining terms, as seen with the concept of "role" categorized as a subclass of "role" confusing. Furthermore, variations in terminology for geographical locations (e.g., "geographical location," "location," "geographic region," etc.) and inconsistent property usage (e.g., "Author," "Composer," etc.) contribute to the challenge.

Additionally, inconsistent levels of detail and overspecialization in hierarchies, such as differing recursion depths for concepts like "Human" and "Geographic location," and structural issues like circular dependencies and underutilization of the "sub property of" property further complicate ontology integration. Moreover, the vast number of properties, abstract identifier naming, and determining when to add new subclasses or utilize existing properties presents decision-making challenges. When we merge different sets of LOD from other places or fields, we sometimes fail to connect related items due to these kinds of challenges caused by scattered ontology [14]. This situation is referred to as a "missing link." We need strategies to tackle this issue.

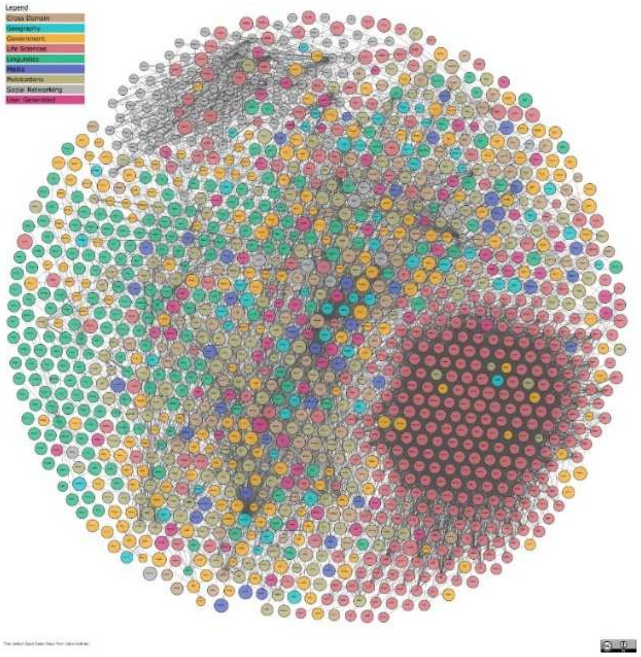


Fig. 2 The Linked Open Data as of March 2024 [15]

This study explores and examines techniques and methodologies that facilitate an ontology mapping process. We propose integrating machine learning techniques that can be applied across various types of LODs. Through importing, extracting, pruning, refining, and evaluating ontologies, our approach offers a solution for the LOD integration system, addressing challenges arising from scattered ontology. The subsequent sections of the paper are structured as follows: Section II reviews two machine learning techniques—feature-based learning and graph-based learning—that can be employed to identify missing links between LOD datasets resulting from scattered ontology. These techniques are extensions introduced in [16]. Section III elaborates extensively on various use case scenarios according to different LOD structures, demonstrating where the machine learning techniques introduced in Section II can be effectively employed. Subsequently, Section IV offers a cumulative discussion of our work and outlines potential avenues for future research.

II. Material and Method

Integrating LODs involves connecting vast amounts of diverse data from various sources characterized by dynamic and heterogeneous structures. LODs are packed with a large amount of ontology, and each LOD uses a different ontology. The complexity of LOD integration is directly proportional to the number of vocabularies and ontologies in the LOD [17].

In general, LOD integration requires the exchange of ontologies, and when ontologies vary, a series of steps are necessary for integration, including link prediction [18]. Link prediction in LOD integration facilitates predictions about missing links between entities in different LOD datasets, even if LOD does not share the same ontology. It involves identifying potential relations that may be logically plausible but cannot be deduced directly from the provided ontology [19].

However, it identifies missing or potential links between entities. The cost of performing link prediction between entities can be prohibitive. Furthermore, while managing small ontologies is relatively straightforward due to the limited number of potential relations, the task becomes more challenging with larger ontologies. Highly skilled domain experts may make mistakes by overlooking certain links or mistakenly adding nonexistent ones [20]. Therefore, we suggest addressing this issue by employing machine learning technologies in LOD integration that can be effective and possibly reduce costs.

Individuals typically devote more time to reformatting and integrating data than the analytical process itself [21]. Our proposed approach involves predicting the connection between non-linked entities during the integration of LOD, even when missing links arise from scattered ontologies. In this section, we review the machine learning technologies that can be used to predict links for LOD integration using similarity-based heuristics: feature-based learning and graph-based learning. These methods can be implemented using relationships between entities based on their ontology. We believe that the capability to effectively predict missing links between LOD datasets significantly impacts the dissemination of information.

A. Link Prediction with Feature-based Learning

Feature-based learning is a machine learning approach where the model learns patterns or relationships in data by focusing on specific features or attributes. In this approach, the input data is represented as a set of features, and the model learns to make predictions or classifications based on these features. To apply feature-based learning to identify missing links among RDF triples in LOD, we can utilize Word2Vec [22].

The Skip-Gram model within Word2Vec has effectively captured semantic relationships and contextual information within large datasets [23]. Skip-Gram is especially adept at embedding rare or infrequent words and is recognized for handling diverse ontologies [22]. This model works by mapping words and phrases from a given vocabulary or corpus into continuous vector spaces, which enables more efficient processing and analysis. Leveraging Skip-Gram, we can predict surrounding entities given a target entity, allowing us to identify missing links between entities.

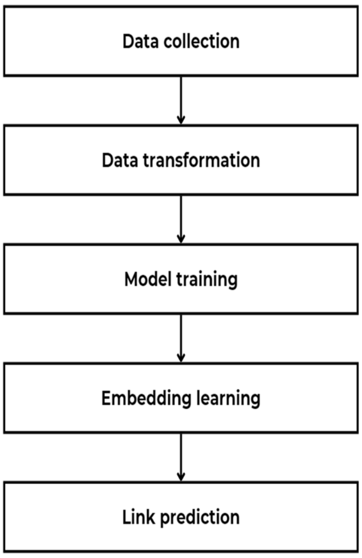


Fig. 3 An overview of a framework for feature-based learning for link prediction in LOD datasets

With consideration to the structure of Linked Data, comprised of RDF triples denoted as Subject - Predicate - Object, we can adapt the framework shown in Fig. 3 to deploy the skip-gram model for link prediction:

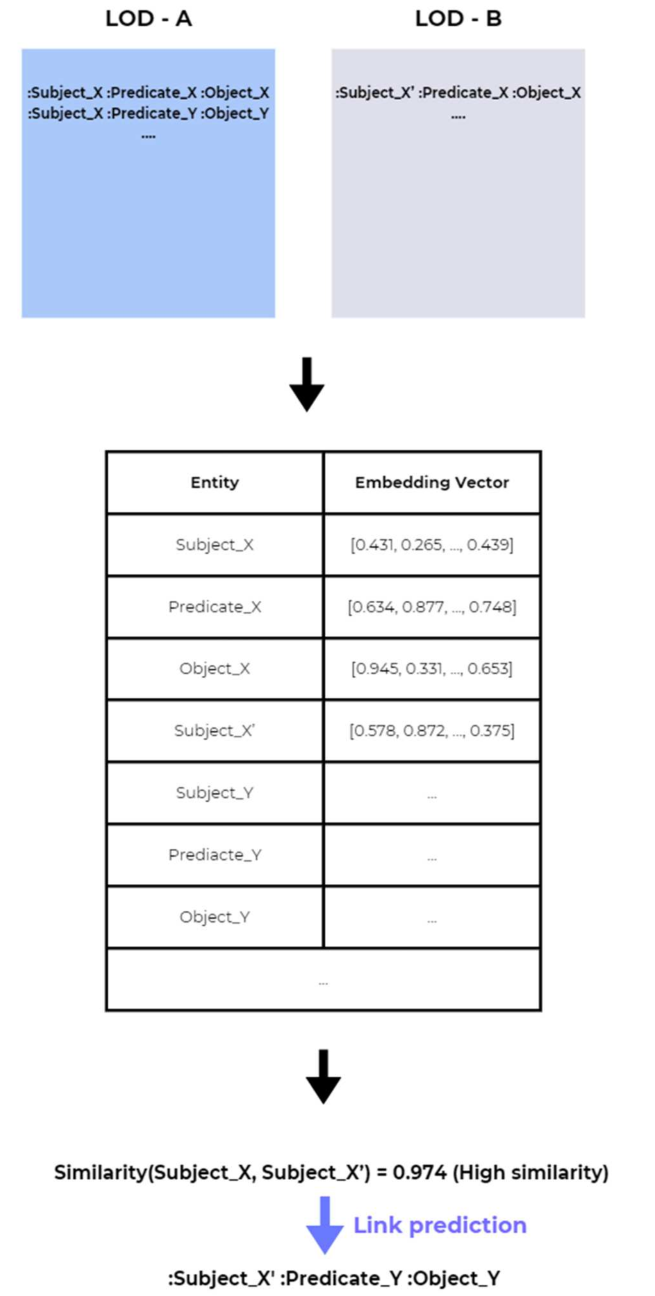
1) Data collection: Collect RDF triple data from linked sources, extracting each Subject, Predicate, and Object.

2) Data transformation: Convert RDF triples to fit the input format of the skip-gram model. Set the Subject as the center word and use the Predicate and Object as context words.

3) Model training: Train the skip-gram model using the transformed data. The model receives the center word "Subject" as input and predicts the "Predicate" and "Object" as surrounding words. During training, the skip-gram model aims to minimize the loss function by predicting surrounding words based on the input center word “Subject,” with the goal of reducing the disparity between expected and actual context words. Negative sampling is utilized as a loss function, emphasizing the understanding of the relationship between the “Subject” and its surrounding words “Predicate” and “Object,” thereby maximizing the probability between correct “Predicate” and “Object” and a subset of negative samples.

4) Embedding learning: The skip-gram model learns embeddings (vector representations) for "Subject" based on the given RDF triple data, understanding the relevance of "Subject" to surrounding words.

5) Link prediction: Utilize the trained embeddings to compute similarities with other RDF triples and discover other words (e.g., “Predicate” and “Object”) related to the "Subject."


LOD - A
:Subject_X :Predicate_X :Object_X
:Subject_X :Predicate_Y :Object_Y
....
LOD - B
:Subject_X' :Predicate_X :Object_X
....

Entity	Embedding Vector
Subject_X	[0.431, 0.265, ..., 0.439]
Predicate_X	[0.634, 0.877, ..., 0.748]
Object_X	[0.945, 0.331, ..., 0.653]
Subject_X'	[0.578, 0.872, ..., 0.375]
Subject_Y	...
Prediacte_Y	...
Object_Y	...
...	

Similarity(Subject_X, Subject_X') = 0.974 (High similarity)
Link prediction
:Subject_X' :Predicate_Y :Object_Y

Fig. 4 Illustrative Example: Link Prediction Using Skip-gram in LOD Integration

Fig. 4 provides a straightforward illustration of link prediction using skip-gram. Once the skip-gram model is trained, it generates word embeddings for entities within LOD. We can anticipate missing connections using these embeddings and the trained skip-gram model. In this example, the prediction of missing links between "*Subject_X'*" and "*Predicate_Y Object_Y*" is made based on the similarity scores calculated.

The skip-gram model learns the association between "Subject" and surrounding words, "Predicate" and "Object," generating embeddings that encapsulate semantic similarities. Through negative sampling, the model strives to maximize the disparity in probability between surrounding words and negative samples while training word embeddings [22].

B. *Link Prediction with Graph-based learning*

In recent years, graph-based machine learning has surged in popularity, attributed to its ability to handle intricate data structures [24]. In the realm of graph structures, link prediction endeavors to forecast future connections between nodes, with edges symbolizing interactions among them [25]. Within the domain of LOD, subjects and objects correspond to nodes, while predicates can be likened to edges that link subjects to objects. Therefore, we can represent LOD as graphs and employ link prediction methods.

This approach can be a scalable tool for link prediction, enhancing the error-prone LOD integration process. We can explicitly employ subgraph-based methods to apply graph-based learning to identify missing links among RDF triples in LOD. Subgraph-based methods involve extracting local subgraphs surrounding each target link and learning subgraph representations through Graph Neural Networks (GNNs) for link prediction. In other words, the approach entails extracting subgraphs surrounding each target link and then classifying whether these subgraphs contain missing links [26].

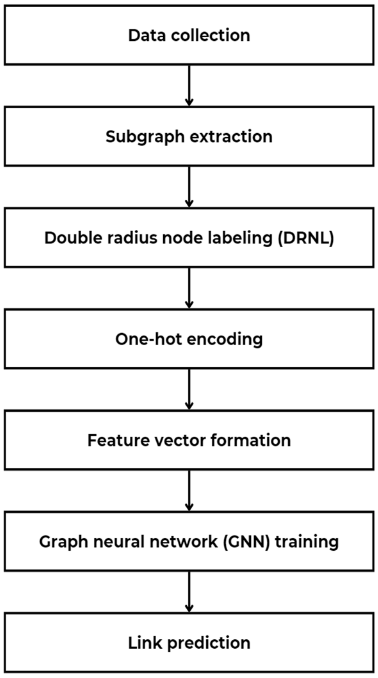


Fig. 5 An overview of a framework for graph-based learning for link prediction in LOD datasets

We can adopt the framework [27] shown in Fig. 5 to deploy the subgraph-based method for link prediction:

1) Data collection: Gather RDF triple data from the LOD sources. Each RDF triple consists of a subject, a predicate, and an object, representing a relationship between entities.

2) Subgraph extraction: For each target link (e.g., subject-predicate-object triple), extract a local subgraph from the LOD dataset. This subgraph should include neighboring entities and their relationships surrounding the target link.

3) Double radius node labeling (DRNL): Apply DRNL to each extracted subgraph to assign integer labels to nodes based on their relative positions and distances to the target entities within the subgraph.

4) One-hot encoding: Convert the DRNL labels into one-hot encoding vectors, where each node's label is represented as a binary vector.

5) Feature vector formation: Concatenate the one-hot encoded labels with the original node features (e.g., entity attributes or properties) to create new feature vectors for each node in the subgraph.

6) Graph neural network (GNN) training: Utilize the labeled subgraphs and their associated feature vectors to train a GNN using supervised learning techniques. The GNN learns to predict missing links between entities based on the structural and feature information encoded in the subgraphs.

7) Link prediction: Once the GNN is trained, it is applied to predict missing links between entities in the LOD dataset. The GNN assigns likelihood scores to potential links, indicating the probability of a link between pairs of entities within the LOD dataset. These predicted links represent potential relationships not explicitly defined in the original LOD data but are inferred based on the learned patterns and associations within the dataset.

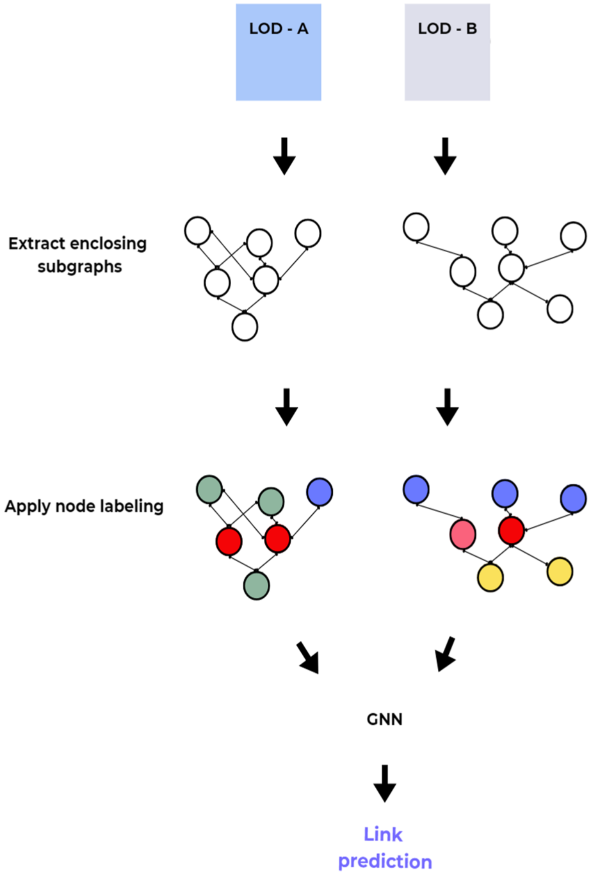


Fig. 6 Illustrative Example: Link Prediction Using subgraph -based method in LOD Integration

Fig. 6 illustrates the extraction and prediction process based on subgraphs surrounding target links. Nodes with different roles are distinguished by applying node labeling within these subgraphs, and labeled subgraphs are supplied to GNNs for supervised learning-based link prediction.

III. Results and discussion

In this section, we delve into real-world scenarios illustrating the practical application of machine learning techniques introduced in Section 2 on LOD datasets. We specifically concentrate on their effectiveness in tackling present missing link scenarios [13], which we briefly introduced in Section I, using link prediction methodologies. We show both skip-gram and subgraph-based learning methods offer practical strategies for predicting links between different LOD datasets with varying ontologies, addressing challenges such as ambiguous concepts, inconsistent properties, and variations in terminology. By leveraging the semantic and structural information encoded in these methods, we can enhance the integration and interoperability of heterogeneous LOD datasets. For clarity and coherence, we will employ the terms "*LOD_A*" and "*LOD_B*" to denote two distinct LOD datasets intended for integration.

A. Scenario 1: Ambiguous concepts

A scenario arises wherein ambiguous terms are utilized across diverse LOD datasets. In this context, skip-gram helps determine how these words are used in different situations by creating representations that capture their meanings and contexts. These representations help predict connections between things based on similar contexts, regardless of which set of data you're looking at. Similarly, subgraph-based learning looks at the nearby connections and patterns of confusing words to understand what they mean. By studying the patterns in the connections between different things labeled with ambiguous words, DRNL can find common themes or groups, showing us what these ambiguous words are all about.

For instance, in *LOD_A*, "Scientist" may be linked with research publications, while in *LOD_B*, it could be associated with academic affiliations, as shown in Fig. 7. Skip-gram adeptly captures such semantic variations, facilitating the prediction of entity links based on shared contexts. Subgraph-based learning encodes the local structural characteristics surrounding ambiguous concepts to discern their intended meanings. It looks closely at the nearby connections and structures around the words "Scientist" to determine their meaning. By examining the neighboring nodes and relationships of instances labeled "Scientist," DRNL can identify common patterns or clusters indicative of the roles or professions associated with these Scientists.

B. Scenario 2: Inconsistent Properties

There are instances of LOD employing inconsistent properties. In such scenarios, both skip-gram and subgraph-based learning offer effective strategies. Skip-gram involves analyzing co-occurrence patterns of entities and properties across diverse datasets to capture the semantics of inconsistent properties. This includes identifying contextual differences in property usage and enabling the prediction of links between entities based on shared properties. Similarly, subgraph-based learning with DRNL examines structural inconsistencies in property usage by analyzing subgraphs surrounding entities with inconsistent properties. By detecting common subgraph patterns across datasets, DRNL infers the underlying semantics of properties and predicts links between entities based on structural similarities.

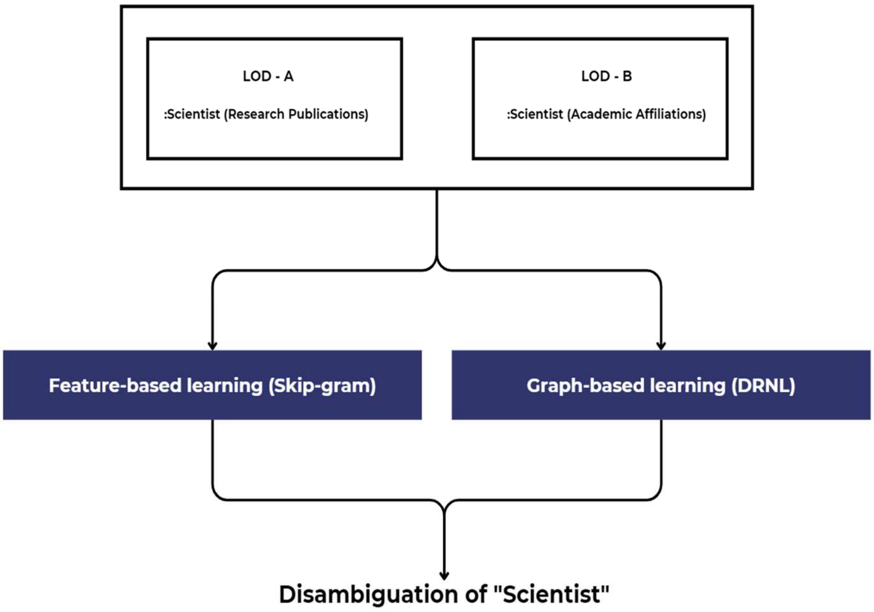

LOD - A
:Scientist (Research Publications)
LOD - B
:Scientist (Academic Affiliations)
Feature-based learning (Skip-gram)
Graph-based learning (DRNL)
Disambiguation of "Scientist"

Fig. 7 An illustrative example of how machine learning techniques work in scenarios with ambiguous concepts

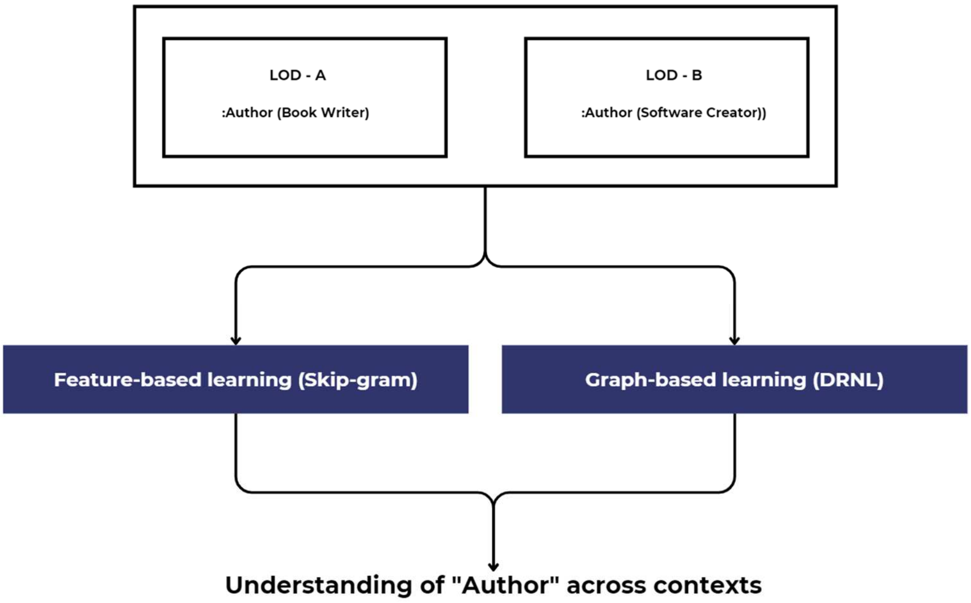

LOD - A
:Author (Book Writer)
LOD - B
:Author (Software Creator))
Feature-based learning (Skip-gram)
Graph-based learning (DRNL)
Understanding of "Author" across contexts

Fig. 8 An illustrative example of how machine learning techniques work in scenarios with inconsistent properties

For instance, if "Author" in *LOD_A* signifies a writer of books while in *LOD_B* it denotes a creator of software, Skip-gram can discern these contextual differences and predict links between entities based on their shared properties as shown in Fig. 8. Similarly, subgraph-based learning can analyze the structural inconsistencies in property usage by analyzing the subgraphs surrounding entities with inconsistent properties. By identifying common subgraph patterns across datasets, DRNL can deduce the underlying semantics of properties and forecast links between entities based on their structural similarities.

C. Scenario 3: Variations in terminology

There are instances where LOD employs diverse terminology to convey similar semantics. Skip-gram focuses on capturing semantic similarities between terms with variations in terminology. By learning representations encapsulating these similarities, Skip-gram facilitates link prediction between entities labeled with different terms yet representing similar concepts. Similarly, subgraph-based learning exploits structural similarities among entities labeled with varying terminology variations. DRNL predicts links between entities labeled with distinct terms but sharing similar structural contexts by analyzing local subgraphs and identifying common patterns corresponding to analogous concepts.

For instance, in *LOD_A*, the term 'geographical location' is used. At the same time, *LOD_B* employs 'geographic region' as shown in Fig. 9. Skip-gram can learn representations, enabling link prediction between entities labeled with different terms yet representing similar concepts. Subgraph-based Learning can capitalize on structural similarities among entities labeled with varying terminology variations by analyzing their local subgraphs. By identifying common subgraph patterns corresponding to analogous concepts, DRNL can predict links between entities labeled with distinct terms yet sharing similar structural contexts.

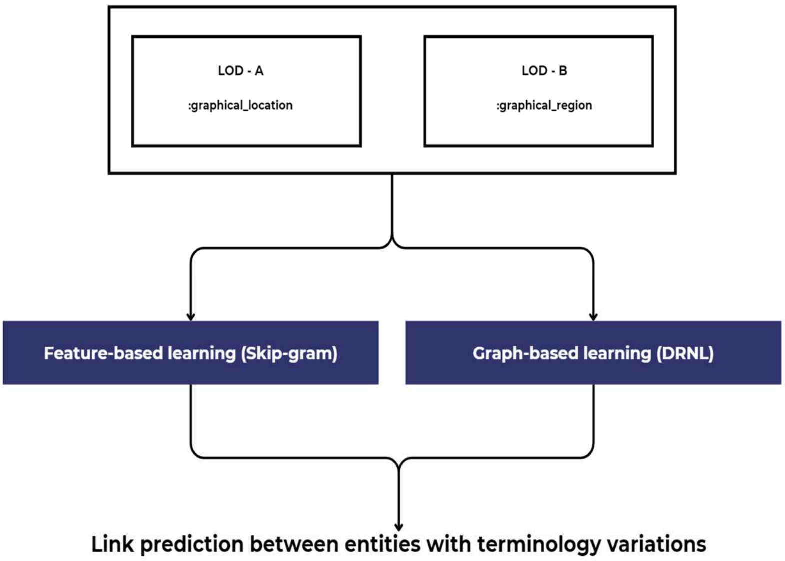

LOD - A
:graphical_location
LOD - B
:graphical_region
Feature-based learning (Skip-gram)
Graph-based learning (DRNL)
Link prediction between entities with terminology variations

Fig. 9 An illustrative example of how machine learning techniques work in scenarios with variations in terminology

IV. Conclusion

This paper has comprehensively explored the challenges posed by scattered ontology management in the context of the LODs and proposed innovative methodologies leveraging machine learning techniques to address these challenges. The integration of LOD datasets from diverse sources, each utilizing different ontologies, often results in difficulties in link prediction between entities, hindering effective data utilization and interoperability [28], [29], [30]. Through the application of feature-based learning and graph-based learning, explicitly using skip-gram and subgraph-based methods with DRNL, we have demonstrated promising strategies for predicting missing links and enhancing LOD integration.

The efficacy of these methodologies has been illustrated through various real-world scenarios, showcasing their ability to handle ambiguous concepts, inconsistent properties, and variations in terminology across different LOD datasets. Skip-gram and subgraph-based learning techniques offer complementary approaches, focusing on capturing semantic similarities and structural patterns to predict missing links between entities. These machine-learning techniques facilitate more efficient and accurate LOD integration processes by encoding semantic and structural information embedded within LOD datasets.

The proposed methodologies hold significant potential for advancing the interoperability and utilization of LODs across diverse domains. Future research endeavors may explore further optimization and refinement of these techniques, considering the evolving landscape of LODs and emerging challenges in ontology management. Additionally, integrating other machine learning approaches and hybrid models could offer enhanced capabilities for addressing complex LOD integration scenarios, ultimately contributing to realizing more interconnected and accessible LODs.

References

[1] Mohammed, W.M.S. and Jumaa, A.K, A survey on using semantic web with big data: challenges and issues. Harbin Gongye Daxue Xuebao/Journal of Harbin Institute of Technology. 54. 93-103, 2022.

[2] C. Bizer, T. Heath, and T. Berners-Lee, "Linked Data - The Story So Far," International Journal on Semantic Web and Information Systems, vol. 5, no. 3, pp. 1–22, Jul. 2009, doi:10.4018/jswis.2009081901.

[3] A. Scherp, G. Groener, P. Škoda, K. Hose, M.-E. Vidal, Semantic web: Past, present, and future, Trans. Graph Data Knowl, 2024, doi:10.4230/ TGDK.2.1.3.

[4] Hitzler, P.; Kroetzsch, M.; Parsia, B.; Patel-Schneider, P.; Rudolph, S. OWL Web Ontol. Lang. Primer (Second Edition). W3C Recomm. 2009, 27, 123.

[5] World Wide Web Consortium. Linked Data - Connect Distributed Data across the Web., 2013. Retrieved from https://www.w3.org/2013/data/

[6] K. Menzel, S. Törmä, K. Markku, K. Tsatsakis, A. Hryshchenko, and M. N. Lucky, "Linked Data and Ontologies for Semantic Interoperability," SpringerBriefs in Applied Sciences and Technology, pp. 17–28, 2022, doi: 10.1007/978-3-031-04670-4_2.

[7] Djebri, A.E.A.: Uncertainty management for linked data reliability on the Semantic Web. Ph.D. thesis, Université Côte D'Azur, 2022. Retrieved from https://hal.archives-ouvertes.fr/tel-03679118

[8] C. Zinke-Wehlmann et al., "Linked Data and Metadata," Big Data in Bioeconomy, pp. 79–90, 2021, doi: 10.1007/978-3-030-71069-9_7.

[9] J. Holze and Milan, "Home," DBpedia Association, Aug. 9, 2022. [Online]. Available: https://www.dbpedia.org/

[10] "Wikidata: Introduction," Wikidata. [Online]. Available: https://www.wikidata.org/wiki/Wikidata:Introduction. [Accessed: Aug. 1, 2024].

[11] "Commons Attribution," [Online]. Available: http://cas.lod-cloud.net/dataset/freebase. [Accessed: Aug. 1, 2024].

[12] "YAGO - Wikipedia," YAGO - Wikipedia. [Online]. Available: https://yago-knowledge.org. [Accessed: Jan. 25, 2024].

[13] A. Westerinen, "Presentation at Wikidata Modeling Days 2023: Modeling and Use of Wikidata in the Semantic Web Community," Dec. 2, 2023. [Online]. Available: https://commons.m.wikimedia.org/wiki/File:Wikidata_Challenges_in_Semantic_Web_Community.pdf. [Accessed: Mar. 10, 2024].

[14] Hassan, B.A, Towards Semantic Web: Challenges and Needs, 2016. ArXiv, abs/2105.02708.

[15] J. P. McCrae, "The Linked Open Data Cloud," [Online]. Available: https://lod-cloud.net/. [Accessed: Aug. 1, 2024].

[16] Y. Kim, J. Lee, S. Oh, J. Kim, J. Mok, C. Noh, and S. Park, "An approach to enrich the structure of the Web of Linked Data," presented at The 7th Int. Conf. Interdisciplinary Research on Computer Science, Psychology, and Education (ICICPE' 2023), 2023.

[17] Nirmaljit Singh and Harmeet Singh, "A Comprehensive Review of Similarity Based Link Prediction Methods for Complex Networks including Computational Biology," Journal of Advanced Zoology, vol. 44, no. S6, pp. 1281–1294, Dec. 2023, doi: 10.17762/jaz.v44is6.2433.

[18] I. Nadim, Y. El Ghayam, and A. Sadiq, "Semantic Annotation of Web of Things Using Entity Linking," International Journal of Business Analytics, vol. 7, no. 4, pp. 1–13, Oct. 2020, doi:10.4018/ijban.2020100101.

[19] Li, N.; Schockaert, S. Ontology Completion Using Graph Convolutional Networks. In Proceedings of the SEMWEB, Auckland, New Zealand, 26–30 October 2019.

[20] S. Mežnar, M. Bevec, N. Lavrač, and B. Škrlj, "Ontology Completion with Graph-Based Machine Learning: A Comprehensive Evaluation," Machine Learning and Knowledge Extraction, vol. 4, no. 4, pp. 1107–1123, Dec. 2022, doi: 10.3390/make4040056.

[21] S. Mhammedi and N. Gherabi, "Heterogeneous Integration of Big Data Using Semantic Web Technologies," Intelligent Systems in Big Data, Semantic Web and Machine Learning, pp. 167–177, 2021, doi:10.1007/978-3-030-72588-4_12.

[22] T. Mikolov, K. Chen, G. Corrado and J. Dean, "Efficient estimation of word representations in vector space", *Proc. 1st Int. Conf. Learning Representations*, 2013.

[23] J. Samuels and J. Samuels, "Exploring novel approaches in word embeddings: A comparative analysis of Word2Vec Skip-Gram models," *ResearchGate*, 2024.

[24] L. da F. Costa et al., "Analyzing and modeling real-world phenomena with complex networks: a survey of applications," Advances in Physics, vol. 60, no. 3, pp. 329–412, Jun. 2011, doi:10.1080/00018732.2011.572452.

[25] T. J. Lakshmi and S. D. Bhavani, "Link prediction approach to recommender systems," Computing, vol. 106, no. 7, pp. 2157–2183, Oct. 2023, doi: 10.1007/s00607-023-01227-0.

[26] M. Zhang, "Graph Neural Networks: Link Prediction," Graph Neural Networks: Foundations, Frontiers, and Applications, pp. 195–223, 2022, doi: 10.1007/978-981-16-6054-2_10.

[27] M. Zhang and Y. Chen, "Link prediction based on graph neural networks", Proc. NeurIPS, pp. 5165-5175, 2018.

[28] N. Noy, Y. Gao, A. Jain, A. Narayanan, A. Patterson, and J. Taylor, "Industry-scale Knowledge Graphs: Lessons and Challenges," Queue, vol. 17, no. 2, pp. 48–75, Apr. 2019, doi: 10.1145/3329781.3332266.

[29] H. Paulheim, "Knowledge graph refinement: A survey of approaches and evaluation methods," Semantic Web, vol. 8, no. 3, pp. 489–508, Dec. 2016, doi: 10.3233/sw-160218.

[30] A. Hogan et al., Knowledge Graphs. Springer International Publishing, 2022. doi: 10.1007/978-3-031-01918-0.