

Exploring the Linked Data Life Cycle: Challenges, Innovations, and Future Directions

Group G05
Gabriel Urbaitis, Nathaniel Filer

04/30/2024

1 Introduction

The semantic web is the future of the web. Right now, the world wide web consists of many resources available to people through search engines and links, but most of them were not designed to be machine readable, so a computer cannot easily navigate the web and read its resources. The vision of the semantic web is to have a machine readable web. In order to do that, the resources on the web must be findable, have descriptive metadata, and be linked to other resources so computers can use semantic queries to use them. How is this done? Right now, the transition to the semantic web is being pushed forward by the transformation of normal web resources into linked data resources, which can be read and used by computers.

The life cycle of linked data has several important steps, and this paper explores several of those steps, attempting to identify and describe several challenges, as well as discuss possible solutions or suggestions found in current research in the field. In this paper, the terminology from Rao & Nayak's LinkED methodology [1] will often be used to describe the steps in the life cycle. Section 2 talks about creating linked data, from specifying the dataset, to cleaning the data, to naming, enriching, and converting the data into triples. It discusses some research on why ontologies are so important and several recommendations for making the most of them, as well as potential innovations into data conversion using LLMs (large language models). Section 3 discusses the use of linked data including common technologies used, the methodologies for managing it and its applications. An overview of 250 Linked Open Data studies are explored, as well as the WWI Linked Open Data project.

2 Creating Linked Data (Triplification)

Usually linked data is created from existing data in the form of a relational database or documents/dictionaries. Any resource, especially resources that are already on the web, can and

should be converted to linked data. This is done by embedding the metadata for a resource into its webpage if it has one, or creating a file that is machine-readable. These files are often adjacent to the web resource (if it exists); the linked data resource for https://dbpedia.org/page/United_States.html is http://dbpedia.org/page/United_States.rdf. But how is this machine-readable metadata modeled? How is it given meaning so it can be used? In order to understand the answers to these questions, the background of linked open data will be briefly explored, and then the process of creating the data will be described from the beginning.

Linked open data has been a concept almost as long as the semantic web, but research regarding it has grown more recently. “Performing a search in Scopus, which is one of the largest research databases, by using the keywords “linked open data”, 6.104 results were found. By analyzing these results, it can be seen that the year with more publications is 2018 (930 studies). It is well known that the term LOD was mainly coined by Berners-Lee (2006), according to the search performed the first publication in which the term LOD appeared was a conference paper by Rahman, Chakraborty, and Seddiqui, (2003)” [2]. Many of the recent studies on LOD come from various fields wanting better access to data from other researchers in the same field. This goal of interoperability is the driving force behind many of the recent advances in creating, publishing, and maintaining linked open data.

2.1 The RDF Model

Linked data does not use the ER (entity-relationship) model for its conceptual schema because it requires even more flexibility than that model provides. It uses the RDF model. As noted by Tim Berners-Lee, “the RDF model is basically an opening of the ER model to work on the Web” [3]. For its logical schema, instead of the relational model (tables/records), it uses the RDF data model (triples). A triple is made up of a subject (a node), a predicate (a directed edge), and an object (another node). For instance, here are two triples with the subject Albuquerque:

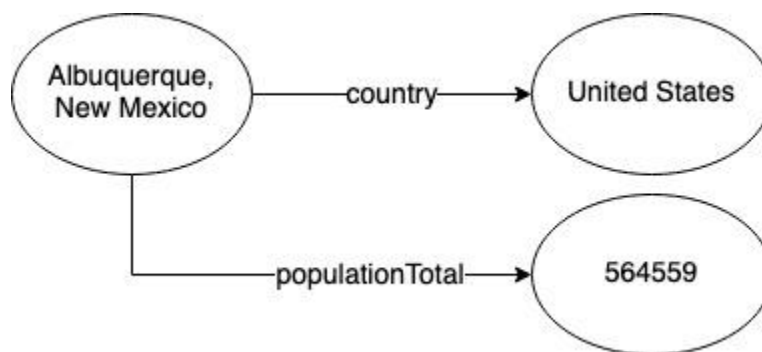


Figure 1

Often, the subjects, predicates, and objects are resources that will need to be referenced by many different triples, so they are referred to by URIs (uniform resource identifiers). These identifiers are unique links that point to a resource—in this case, parsable RDF content in an

RDF serialization format. One format that is frequently used is JSON-LD (which stands for “JavaScript Object Notation for Linked Data”) Part of the reason it’s so widely used is because Google uses it [13]. Below is an example of the same two triples shown above, but expressed in JSON-LD as it would be if it were embedded in an HTML file.

```
<script type="application/ld+json">
{
  "@context": "http://dbpedia.org/ontology/",
  "@id": "http://dbpedia.org/resource/Albuquerque,_New_Mexico",
  "country": {"@id": "http://dbpedia.org/resource/United_States"},
  "populationTotal": 564559
}
</script>
```

Example 1 (from https://dbpedia.org/page/Albuquerque,_New_Mexico)

The “@context” key is not necessary, but in this case it allows the predicates “country” and “populationTotal” to be referred to in a shorter and more readable way instead of using their full URIs. The “@id” key indicates the URI of the subject (“Albuquerque, New Mexico”). The predicate “country” is part of DBPedia’s ontology, and its URI is <http://dbpedia.org/ontology/country>. The object “United States” is referred to by its URI. The object 564559 is not referred to by a URI, since it’s just a number, but its predicate “populationTotal” does specify a range for it (“nonNegativeInteger”). Below is a partial snippet of the DBPedia ontology resource “populationTotal”, serialized in embedded JSON-LD format. Notice the last key-value pair where the range is defined.

```
<script type="application/ld+json">
{
  "@context": {
    "owl": "http://www.w3.org/2002/07/owl#",
    "rdf": "http://www.w3.org/1999/02/22-rdf-syntax-ns#",
    "rdfs": "http://www.w3.org/2000/01/rdf-schema#",
    "xsd": "http://www.w3.org/2001/XMLSchema#"
  },
  "@id": "http://dbpedia.org/ontology/populationTotal",
  "@type": [
    "owl:FunctionalProperty",
    "rdf:Property",
    "owl:DatatypeProperty"
  ],
  "rdfs:range": {"@id", "xsd:nonNegativeInteger"}
}
</script>
```

Example 2 (from <https://dbpedia.org/ontology/populationTotal>)

Here you can also see that multiple contexts (ontologies) can be used to define objects and predicates. Technically, simple strings or other data types that *don’t* point to a resource with a URI can be used as predicates, just like they can be used as values for objects, but URIs should be used wherever possible, otherwise the data won’t be well-linked.

2.2 Preparing the Data

The RDF model is designed to function without a predefined schema, so that the data can grow in any way it needs to. Not only is the schema not predefined; it is also not defined apart from the data. “In Linked Data, the data schema is represented with the data itself. This mechanism of self-describing data contrasts with the relational approach where external documents (e.g., data dictionaries) and diagrams (e.g., entity relationship diagrams, logical schemas) describe the data” [4]. Even so, some level of data modeling will make the process of creating linked datasets much smoother, as well as making the datasets themselves more useful. Because the schema is represented with the data, the first steps to creating linked data are to specify a dataset, make sure the data is consistent (“cleaning”), and organize the data in some way (“extraction”).

While the point of the semantic web is to allow for linked data without scope (all kinds of data linked together), it is currently made up of smaller, domain-specific linked datasets that are then linked together by the digital resources that connect them. Therefore, when looking to publish new linked open data, it is important to pick a dataset in a specific domain to start with [4]. This is similar to the process of picking a mini-world when creating a conceptual schema for a relational database, but it doesn’t need to be exhaustively defined, because whatever scope is picked isn’t ultimately limiting. It’s just for the sake of finding the right starting ontologies for the data.

Once the starting dataset has been specified, the data needs to be validated and “cleaned”. This means data must be combed through and changed if necessary to be in standard, correct formats (dates, phone numbers, and other data types must be consistent) [1]. It is also important to note during this part of the process if there are any other specific types of data, such as various measurements, that need to be expressed in certain units. The units of data should be set to whatever is most useful (usually the standard units in the field the dataset belongs to).

Lastly, before the process of transforming the data into linked data begins, it must be structured in some way. Often, even if the data itself is unstructured, it is already somehow organized into different documents covering different entities or types of entities. If it is not organized at all, it must be organized, at least slightly. Rao refers to this as “extraction” [1]. There are many ways to organize and structure data. One of the most common ways is to start with a relational database containing the data, because there are already tools for converting relational data into linked data, and because relational databases are well organized by nature. In some cases however, that may be entirely unnecessary and take too much time and effort. In those cases, having the data organized into CSV files or documents, each with information about a specific category of data, is sufficient.

2.3 Modeling the Data

Now the objects and subjects in the data must be identified and given names. The Linked method [1] suggests creating URIs in this phase. This process can be a bit tricky, since there are multiple ways to link to RDF resources. The two established approaches are hash URIs and 303 URIs, also sometimes referred to as hash and slash identifiers [5]. During this phase is also a good time to consider what other datasets have resources you want to link to your data, and make sure you know the URIs for those resources as well.

Once the objects and subjects are named, the predicates must be named as well, and additional properties and classes should be chosen to enrich the meaning of the data. If possible, predicates should be named using existing ontologies. Ontologies (in this case, standard vocabularies with established meanings that are accessible by URIs) can be used as semantic schemas to model datasets. These Ontologies aren't necessarily rigid—they can grow and adapt just like the data, but it is important to pick the right ontologies, as you go, for interoperability. For instance, if you use an uncommon ontology or create your own, it may use different terms from the standard vocabularies used in other datasets within your field, which would make linking the datasets difficult.

Some of the current work being done in modeling linked data is developing common vocabularies for different fields. People are also spreading awareness on how to find standard vocabularies and use them. This is crucial to the improvement of the semantic web—interoperability is the whole point of linked data. If people don't use standard vocabularies, it will be difficult to link their data in a meaningful way. This is why the data should be modeled in the first place—to be able to link to as much existing, high-quality linked data as possible.

One question encountered here is whether to use only one controlled vocabulary, or multiple vocabularies. There are obvious benefits (as mentioned, interoperability), but for datasets that cover diverse data from different domains, it can be difficult to find vocabularies that fit. “A vocabulary that attempts to be universal in both scope and detail can become bloated and even run the risk of resembling an aggregated list of uncontrolled terms” [6]. And, if trying to fit diverse data to a vocabulary that doesn't cover it (many resources without good matches in the vocabulary), then “inadequate subject representation” [6] is encountered. This means that either resources must be described imperfectly, uncontrolled terms must be used, or other vocabularies must be found. We suggest using combinations of vocabularies, one for each domain touched upon by the dataset, especially since the number of controlled vocabularies available for use has grown significantly over the years. In fact, “most data providers use between 3 and 8 vocabularies” [7]. Also, if a number of existing vocabularies are used but there is still further need for terms that haven't been defined, a data provider can create their own vocabulary just for the terms not contained in the existing ontologies. Most established vocabularies can be found at the “linked open vocabularies” website.

Another suggestion found in research is to use special terms to map one vocabulary to another when two datasets use different vocabularies, but need to be linked. “Whilst there is little chance of obtaining domain-wide agreement on common vocabularies, the prospects for large-scale interoperability can be substantially improved by creating alignment links from datasets out to external reference resources” [8]. Thankfully, this can be done in a number of different ways. There are now several existing ontologies developed for mapping other vocabularies onto each other. “Mapping properties (SKOS 2012; ISO 25964-2 2013) and also OWL property “sameAs” (Ontology Web Language Overview 2004) enable the mapping of concepts in a data resource to concepts in another” [9]. With the amount of standard vocabularies that exist now, and the ability to map terms when different vocabularies are used for similar data, interoperability is entirely within reach when modeling a new linked open dataset.

2.5 Converting the Data to Triples

Once the data has been prepared and modeled, it needs to be converted into triples. There are existing tools for converting data to RDF triples, but most of them use a mapping language (R2RML, RML, YARRRML, or YAML can be used), which can take a while to learn for those who don’t already know them. Another approach is to create templates in an RDF serialization language, such as JSON-LD or Turtle, and fill in the templates manually.

This seems to be an area where AI (specifically LLMs) could be of use. As an experiment, we created a paragraph with data about a fictitious city called “Pendrit” and asked ChatGPT to create a list of RDF triples from it. That worked pretty well, but it didn’t use URIs or common vocabularies, of course. Asking ChatGPT to write a JSON-LD RDF data serialization using the schema from schema.org (the vocabulary developed for use by Google and Microsoft) worked well too, but sometimes ChatGPT would create URIs to terms that didn’t exist in the vocabulary—it made mistakes. In our brief, naive experiment, in order to get ChatGPT to work without mistakes, we had to create a *limited* template and ask ChatGPT to fill in the placeholders with information from the paragraph. Below is the prompt that was used.

Given the following JSON-LD template and the paragraph after it, use the information from the paragraph to replace every possible set of <> placeholders and the descriptions they contain. Unless otherwise specified, when filling in the template, use underscores instead of spaces, and only fill in information that is explicitly stated—don’t assume anything. If there is a placeholder in the template for which there is no matching information in the paragraph, remove that line from the JSON-LD script.

The JSON-LD template:

```
{
  "@context": {
    "dbr": "http://dbpedia.org/resource/",
    "dbo": "http://dbpedia.org/ontology/",
    "dbp": "http://dbpedia.org/property/"
  },
  "@id": "dbr:<Name of city>",
```

```

"@type": "dbo:city",
"dbo:country": "dbr:<Name of country that contains city>",
"dbo:region": "dbr:<Name of region that contains city>",
"dbp:location": [<Comma separated list of areas that contain the city, each
list item should be formatted: "dbr:<>">],
"dbp:name": "<Name of city with spaces instead of underscores>",
"dbo:namedAfter": "dbr:<Name of person the city was named after>",
"dbp:nicknames": [<Comma separated list of city nicknames and abbreviations
without underscores, each list item should be formatted: "<>">],
"dbo:populationMetro": <Population of metropolitan area>,
"dbo:populationTotal": <Population of city>,
"dbo:populationTotalRanking": [<City's population ranking in region>,
<City's population ranking in country>],
"dbp:establishedDate": <Date city was established, in format YYYY-MM-DD>,
"dbo:founder": "dbr:<Name of founder of city>"
}

```

The paragraph:

Pendrit, abbreviated PDT, is the capital city of the Penwald region in the country of Keloth. With 56,762 registered residents, it is the most populated city in Penwald and the 2nd most populated in Keloth. It is located just north of Kamris lake, in the Great Plains of Tilar. The surrounding area (which, including the city, contains 94,739 people) is used as farmland by many, and Pendrit's primary industry is agriculture. Even so, the city is also home to the headquarters of the guild of war, the survivalists guild, and the guild of knowledge, making it a hub for all sorts of activity. It was founded in the year 4062 by Hadof Penwald as part of the Geldrian initiative to settle the Kelothian continent.

Example 3 (the prompt that was used for ChatGPT)

```

{
  "@context": {
    "dbr": "http://dbpedia.org/resource/",
    "dbo": "http://dbpedia.org/ontology/",
    "dbp": "http://dbpedia.org/property/"
  },
  "@id": "dbr:Pendrit",
  "@type": "dbo:city",
  "dbo:country": "dbr:Keloth",
  "dbo:region": "dbr:Penwald",
  "dbp:location": ["dbr:Kamris_lake", "dbr:Great_Plains_of_Tilar"],
  "dbp:name": "Pendrit",
  "dbo:namedAfter": "dbr:Hadof_Penwald",
  "dbp:nicknames": ["PDT"],
  "dbo:populationMetro": 94739,
  "dbo:populationTotal": 56762,
  "dbo:populationTotalRanking": [1, 2],
  "dbp:establishedDate": "4062-01-01",
  "dbo:founder": "dbr:Hadof_Penwald"
}

```

Example 4 (the code that was returned by ChatGPT)

Even though work by ChatGPT needs to be proof-read, this could still be much faster than filling in a template manually, especially if several templates for different types of resources (people, locations, objects, etc.) are created and reused for many specific resources. The other benefit to this method (or variations on it), even though it might take longer than some more automatic tools for converting to RDF triples, is that it requires templates to be made, which requires the data model to be thought about very carefully. In the end, the extra time spent might be worth the increased quality of the data schema.

After thinking about this simple little experiment, another question is encountered: is there any research being done into using AI for open information extraction (a way to get structured data from unstructured data)? McCusker [10] describes an approach to Knowledge Graph generation through prompt engineering using OpenAI's text-davinci-003 model. This directive prompt was given in order to define how to process the unstructured text prompts that followed:

Given a prompt, extrapolate as many relationships as possible from it and provide a list of many fine-grained simple links. If a link is a relationship, provide [ENTITY 1, RELATIONSHIP, ENTITY 2]. Use modeling approaches as similar as possible to Wikidata. Links must correspond to Wikidata properties. Links must be the simplest possible relationships between as many entities as possible. The relationship is directed, so the order matters. If the link contains entities, it should be broken apart into multiple entities and relationships. Entities should not contain 'and', but should be broken into the smallest possible groupings. If the link is a relationship between an entity and a value, the values should have their data type after them, in the form [ENTITY 1, RELATIONSHIP, VALUE, TYPE]. Roles should be expressed as relationships between entities. (McCusker [10], p. 8)

The prompt then generates data in the format they desire (a JSON list of lists). Wikidata-based entity linking is employed for both RDF construction and to improve triple quality. A tool called spaCy takes the subject, predicate and object identified by the text-davinci-003 model, and performs named entity recognition. It matches the subject and object against their entity index based off of a 2020 snapshot of Wikidata. After potential matches are retrieved from the index, the candidate with the smallest edit distance is selected as the most likely match. The process is intended to validate the triple to the best ability of the Wikidata database.

The model, OKE-GPT (Open Knowledge Extraction - Generative Pre-trained Transformer), outperformed a model commonly used for knowledge graph construction, OpenIE 4 (Open Information Extraction 4) in linkability for subjects .793 to .532, .802 to .587 on predicates, .670 to .123 on objects and .437 to .031 for triples. The difference was attributed to a focus on subject/object symmetry in OKE-GPT, which makes reuse of generated entities in other statements easier when a new entity is introduced to the knowledge graph.

It is fascinating that, with the right, clearly worded prompt, an LLM can execute even complicated tasks well. In fact, after seeing the research that was done here, it is not

inconceivable that the right model could produce serialized RDF data, using correct formatting for URIs and words from established vocabularies for predicates, types, and classes, without even needing a template, as long as it is provided with the right information. The model could be fed the text from a webpage and return the linked data, which could then be embedded into the html as linked open data, and the whole process would be significantly shortened.

3 Using Linked Data

The use of linked data may be explored through several lenses: the technologies used, the methodologies for managing it and its applications. We focus here on Linked Open Data (LOD), data that is available to the public. Avila-Garzon [2] performed a systematic literature review on LOD, reviewing 250 articles to provide a general overview of applications, technologies, and methodologies for LOD. We review Avila-Garzon's findings here, and examine one of the studies mentioned in the findings, Lindquist, Dulock, Törnroos, Hyvönen, and Mäkelä's WWI LOD project [11].

According to Avila-Garzon [2], In a wide number of the studies selected, DBpedia was the tool most used. It is a knowledge base that has an extraction framework to get structured information from Wikipedia and turn it into a machine readable format. It is the source that most other data sources are connected with. The next two most used technologies were RDF, the machine readable format, and SPARQL, a language used to make queries on the RDF data sets. The technologies are supported by infrastructures based on servers such as Virtuoso or D2RQ. Other notable technologies are SKOS, the specifications and standards for the use of Knowledge Organization Systems, GeoNames, a dataset used for applications related with geographical points, and Europeana, the European digital library. Each was used by at least 5 studies, with DBpedia, RDF and SPARQL leading the way with 54, 35 and 22 studies respectively.

In addition, Avila-Garzon examined 8 studies focusing on methodologies for managing LOD, compared in the table below.

Table 4. Methodologies for LOD

Study	Interlinking	Annotation	Publication	Retrieval	Content generation	Transformation	Storage	Visualization
(Beneventano, Bergamaschi, Sorrentino, Vincini, and Benedetti, 2015)	x	x	x					
(Torre-Bastida, Villar-Rodriguez, Del Ser, and Gil-Lopez, 2015)	x			x				
(Anibaldi, Jaques, Celli, Stellato, and Keizer, 2015)	x		x			x		
(Kawamura and Ohsuga, 2015)	x		x		x			
(Braşoveanu et al., 2017)	x							x
(Rao and Nayak, 2017)	x		x	x	x	x	x	x
(Selvam, Balakrishnan, and Ramakrishnan, 2018)	x			x				
(Alobaidi, Malik, and Sabra, 2018)		x			x			

According to Avila-Garzon [2], None of the methodologies involves a complete process for generating and publishing LOD resources. The LinkED Methodology for Publishing Linked Enterprise data [1] introduces a methodology for publishing Linked Data from Enterprises which can be applied to Open Data contexts. It defines Interlinking as “establishing semantic links between the source dataset and other potential target data sets on the World Wide Web”. It explains Publication of Data “in machine-readable RDF format [which] is linked to other data sources on the Web through URI links, thereby contributing towards a giant global data space.” The context of Retrieval in the study was the process of getting “RDF triples from the storage [which] can be performed in two ways: a) through the SPARQL endpoint, which allows the dataset to be queried through SPARQL queries or b) through the RDF dump, where the entire dataset is exposed as a set of triples under 'creative commons' license for open downloading and subsequent usage.” Content generation took on two forms in the study, link generation, and link statistics. Link generation was done through the SILK framework using a pre-matching algorithm to compare different RDF datasets and employing BM25, a method using term frequency and document length normalization to determine the relevance of a document to a given query. Link statistics were described as “[showcasing] the total number of links generated between the source and target data sources.” Transformation was the process of turning raw data into RDF Format. It includes extraction from source into open structured formats like CSV files or SQL databases, naming the unique resources within the extracted data, enriching them with an ontology, associating with the ontologies classes and properties, and conversion of the raw data into RDF based on a serialization format like RDF/XML, Turtle, N-Triples and N3. The Storage typically used is a specialized database called a “triple store.” Visualization is termed as the presentation of RDF data in HTML, JSON, CSV or XML formats, but as these are primarily for

machines, it is extended to its showcasing via charts, graphs and maps, with potential dataset statistic display.

Annotation is not described in Rao and Nayak's study, but it is used in Beneventano, Bergamaschi, Sorrentino, Vincini, & Benedetti's study on Semantic Annotation of the CEREALAB Database [12]. They describe semantic annotation as "the explicit association of one or more meanings to schema element labels (classes and attributes names)." In their process, they use a Data Integration System called MOMIS to automatically annotate the CEREAL LAB Database with the AGROVOC linked dataset.

In the comparison, Interlinking was the most popular methodology described, with all but one study mentioning it. Half or close to half the studies looked at Publication, Retrieval and Content generation, and only a quarter of the studies looked at Annotation, Transformation and Visualization. Rao and Nayak [1] were the only ones to discuss Storage in the comparison. The comparison of methodologies was only 8 studies in size so a larger sample might change the trends a bit. A main drawback identified was that "not all the steps or phases in a LOD process are considered in an integrated methodology," so therefore, "future research should focus on the development of validated standard methodologies for managing LOD."

Of the 250 articles reviewed, Biology and Social Sciences were the main areas of application with about 20% of the studies each. Biology topics included Biology in general, agriculture, biomedicine, biodiversity, chemistry, climate, ecology, and electrical energy. Social Science topics included Cultural heritage, disabilities, event modeling, geography, government data, linguistics, meteorology, museums, paleontology, piracy, public procurement, social media, spatial data, transport, and tourism. Other areas of application were Libraries, Semantic Web, Research, Education and Management from 8.4% to 2.8% with the remaining 34.8% being classified as "Other," each with less than 2.8%. Some of the Topics included were Natural language processing, Tools focused on managing research data, Tools focused on managing academic data, Accounting, Arts, Augmented Reality, Census data, and Music.

To take a look at one of these examples, one can examine the WWI Linked Open Data Project (WWI LOD) (Lindquist, Dulock, Törnroos, Hyvönen, and Mäkelä [11]), one of many initiatives by libraries around the globe undertaken to transform their bibliographic data to machine readable formats such as RDF. The project focused on linking data related to the civilian experience during World War I in occupied Belgium and France. The primary data sources were from the University of Colorado's World War I collection and additional data from external contributors such as the Imperial War Museum and the HISSTAT project.

The main purpose of WWI LOD was to improve the findability and context of historical subjects in digital collections. The project employed various technologies and methodologies including RDF, specialized vocabularies for semantic linking, and tools like MarcEdit for metadata conversion. DBpedia provided additional contextual information that helped to enrich

the primary source data, making it more accessible and informative for users. The links to DBpedia allowed for a richer contextualization of the sources by connecting them to a broader base of knowledge about people, places, events, and topics associated with WWI. GeoNames was leveraged to resolve and standardize geographical names found in their primary source data. By linking related concepts across WWI datasets using specialized vocabularies and enabling semantically rich services, the authors hope to empower users to find and use online primary sources efficiently and effectively.

Their future goals include adding annotations using tools such as DBpedia Spotlight and Wikimeta, and collaborating with other WWI data projects in Australia, New Zealand, Canada, Great Britain, and the European Union to link their data and contribute collectively to a WWI node in the LOD cloud. The LOD Cloud is a large-scale visualization and infrastructure that demonstrates the implementation and interconnection of various datasets published according to Linked Open Data principles. It represents an ecosystem where datasets are openly available and semantically linked to one another, enabling them to be queried in a unified manner across diverse sources. Finally, they mention their most important goal as creating a user interface, an often overlooked crucial step in making linked open data projects more accessible to people outside the information sciences.

4 Conclusions

One of the most important parts of the linked data life cycle is the enrichment of the data using linked open vocabularies. If the modeling of the data is done correctly, it will carry more meaning and be more useful to both people and machines. It will also be well-linked, and have more potential to contribute to the success of other datasets. Additionally, if the data is modeled using the right ontologies, it will be easier to convert into triples efficiently, potentially by automatic tools or even LLMs with the help of prompt engineering. Two ways to make sure the data is well-modeled are to use multiple vocabularies when necessary, and to map your vocabularies to other ontologies that are used by similar datasets.

The most used technologies are DBpedia, RDF and SPARQL. In regards to methodologies, most of the studies apply their own steps for manipulating LOD and more research is needed to define a standardized methodology for managing LOD. Applications of LOD are led by Biology and Social Sciences with 20% each of studies observed, followed by Libraries, Semantic Web, Research, Education and Management making up another 25% total, and the remaining 35% being made of miscellaneous applications with less than 2.8% of studies observed. The WWI LOD project highlights common elements of Linked Open Data projects, using specialized vocabularies and ontologies to link related concepts within the datasets, linking internal data to external data sources like DBpedia and Geonames, and its goal of contributing back to the LOD cloud with a WWI node, so that other projects may use its work much like it used the DBpedia node.

References

- [1] Rao, S. S., & Nayak, A. (2017). LinkED: A Novel Methodology for Publishing Linked Enterprise Data. *Journal of Computing and Information Technology*, 25(3), 191-209. DOI: 10.20532/cit.2017.1003477
- [2] Avila-Garzon, Cecilia. "Applications, Methodologies, and Technologies for Linked Open Data: A Systematic Literature Review." *International Journal on Semantic Web and Information Systems*, vol. 16, no. 3, July 2020, p. NA. Gale Academic OneFile, link.gale.com/apps/doc/A759324378/AONE?u=albu78484&sid=ebsco&xid=973663e4. Accessed 24 Apr. 2024.
- [3] <https://www.w3.org/DesignIssues/RDB-RDF.html>
- [4] <https://www.w3.org/TR/ld-bp/>
- [5] Zhang, S., Benis, N., & Cornet, R. (2023). Assessing Resolvability, Parsability, and Consistency of RDF Resources: A Use Case in Rare Diseases. *Journal of Biomedical Semantics*, 14(19), 1-14. <https://doi.org/10.1186/s13326-023-00299-3>
- [6] Radio, E., & Hanrath, S. (2016). Measuring the Impact and Effectiveness of Transitioning to a Linked Data Vocabulary. *Journal of Library Metadata*, 16(2), 80–94. <https://doi-org.libproxy.unm.edu/10.1080/19386389.2016.1215734>
- [7] Gottron, T., Knauf, M. & Scherp, A. (2015). Analysis of schema structures in the Linked Open Data graph based on unique subject URIs, pay-level domains, and vocabulary usage. *Distrib Parallel Databases*, 33, 515–553 <https://doi.org/10.1007/s10619-014-7143-0>
- [8] Binding, C., & Tudhope, D. (2016). Improving Interoperability Using Vocabulary Linked Data. *International Journal on Digital Libraries*, 17, 5-21. DOI: 10.1007/s00799-015-0166-y
- [9] Marcondes, C. H. (2022). The Role of Vocabularies in the Age of Data: The Question of Research Data. *Knowledge Organization*, 49(7), 467-482. DOI: 10.5771/0943-7444-2022-7-467
- [10] McCusker, J. (2023). LOKE: Linked Open Knowledge Extraction for Automated Knowledge Graph Construction. CoRR, <https://arxiv.org/abs/2311.09366>
- [11] Lindquist, T., Dulock, M., Törnroos, J., Hyvönen, E., Mäkelä, E. (2013). Using Linked Open Data to Enhance Subject Access in Online Primary Sources. *Cataloging &*

Classification Quarterly, 51(8), 913-928.

<https://www.tandfonline.com/doi/pdf/10.1080/01639374.2013.823583>

- [12] Beneventano, D., Bergamaschi, S., Sorrentino, S., Vincini, M., & Benedetti, F. (2015). Semantic annotation of the CEREALAB database by the AGROVOC linked dataset. *Ecological Informatics*, 26(P2), 119–126.

<https://www.sciencedirect.com/science/article/abs/pii/S1574954114000843?via%3Dihub>

- [13] Baskauf, S. J. (2021). Having Your Cake and Eating It Too: JSON-LD as an RDF serialization format. *Biodiversity Information Science & Standards*, 1–2.

<https://doi-org.libproxy.unm.edu/10.3897/biss.5.74266>