CIT. Journal of Computing and Information Technology, Vol. 25, No. 3, September 2017, 191–209
doi: 10.20532/cit.2017.1003477

LinkED: A Novel Methodology for Publishing Linked Enterprise Data

Shreyas Suresh Rao and Ashalatha Nayak

Department of Computer Science and Engineering, Manipal Institute of Technology, Manipal University, India

Semantic Web technologies have redefined and strengthened the Enterprise-Web interoperability over the last decade. Linked Open Data (LOD) refers to a set of best practices that empower enterprises to publish and interlink their data using existing ontologies on the World Wide Web. Current research in LOD focuses on expert search, the creation of unified information space and augmentation of core data from an enterprise context. However, existing approaches for publication of enterprise data as LOD are domain-specific, ad-hoc and suffer from lack of uniform representation across domains. The paper proposes a novel methodology called LinkED that contributes towards LOD literature in two ways: (a) streamlines the publishing process through five stages of cleaning, triplification, interlinking, storage and visualization; (b) addresses the latest challenges in LOD publication, namely: inadequate links, inconsistencies in the quality of the dataset and replicability of the LOD publication process. Further, the methodology is demonstrated via the publication of digital repository data as LOD in a university setting, which is evaluated based on two semantic standards: Five-Star model and data quality metrics. Overall, the paper provides a generic LOD publication process that is applicable across various domains such as healthcare, e-governance, banking, and tourism, to name a few.

ACM CCS (2012) Classification: Applied computing → Enterprise computing → Enterprise ontologies, taxonomies and vocabularies

Applied computing → Enterprise computing → Enterprise data management

Applied computing → Enterprise computing → Enterprise interoperability → Information integration and interoperability

Software and its engineering → Software organization and properties → Extra-functional properties → Interoperability

Keywords: linked open data, semantic web, enterprise-web interoperability, enterprise data, digital repository

1. Introduction

An enterprise refers to a business organization which creates services for public consumption. Primarily, enterprises are of three kinds:

a) that are driven completely by technology;

b) that employ technology to produce a product or a service;

c) that use technology to benefit the business goals [1].

The third kind of enterprise is of particular interest to this research, which involves domains such as banking, education, supply chain management, healthcare etc. A 2013 study reveals that unstructured and semi-structured data accounts for more than 70 percent of all data in the enterprise [2], while the remaining 30% is structured. However, the majority of this enterprise data exists as "data silos" with a potential for interoperability within and outside the enterprise.

The integration of data silos within enterprises is accomplished by adopting the Enterprise Service Oriented Architecture (ESOA) style, that facilitates application-level and intra-enterprise interoperability [3]. The ESOA is a service-centric umbrella model that encapsulates application logic into Web services. Application level interoperability is achieved using WS-* standards, SOAP (Simple Object Access Protocol) as the messaging format and WSDL (Web Services Description Language) for describing service interfaces [3]. Furthermore, intra-enterprise interoperability is achieved using the Enterprise Service Bus, which acts as the communication system between the interacting enterprise's applications [4].

The World Wide Web can be perceived as a network of Web resources [5], uniquely identified by their URIs (Uniform Resource Identifier). The service-centric ESOA style is not applicable for interoperability with the resources on the Web. To solve this lacuna, enterprises adopted the Enterprise Web-Oriented Architecture (EWOA) style, which is Web-centric and based on the architecture of the Web [6]. Furthermore, the EWOA is based on the HTTP protocol, comprises of RESTful Web services, and supports resource representation formats such as XML, HTML and JSON. Currently, as per literature, the EWOA style is only used by enterprises to retrieve data from Web applications which expose their data via RESTful Web services [6].

In the context of the Web, "linked data" refers to "a set of best practices for exposing, sharing, and connecting pieces of data, information, and knowledge on the Semantic Web" [7]. The data is published in machine-readable RDF (Resource Description Framework) format and is linked to other data sources on the Web through URI links, thereby contributing towards a giant global data space [8]. Typically, the linked data on the Web is also referred to as Linked Open Data (LOD) since the data is available not only to the publishing organization producing the data but also to any other organization or Web users accessing the data across the world. Publishing organizational data openly on the Web provides an opportunity for other organizations, academicians and researchers to analyze and disseminate their work.

The adoption of LOD prescriptions within enterprises empowers the enterprises to publish and interlink their data with the Web of data, using ontologies, thereby strengthening the enterprise-Web interoperability [9].

Although a huge body of research exists with respect to LOD of the Web, the adoption of LOD prescriptions in an enterprise context is limited. Within an enterprise, LOD prescriptions are used primarily for tag-based expert search within enterprise repositories [10], [11], the creation of unified information space [12] and augmentation of core data with additional information [13], [14].

Kaschesky and Selmi [15] proposed a methodology for publishing linked open government data using the R5 framework which stands for Reveal, Refine, Reuse, Release and Run. The approach is specific to government organizations and has the potential to extend for commercial applications. Other methodologies/ approaches include: clinical trials [16], pharmaceutics [17], network science [18], e-governance [19], [20], [21], tourism [22] and meteorological studies [23]. However, these approaches are domain-specific [16] – [19], ad-hoc [15] and therefore lack uniform representation across domains.

Hogan *et al.* [24] surveyed 38 LOD dataset papers published over 4 years from 2012 to 2016 and identified three unresolved and recurrent issues in LOD datasets, namely: existence of inadequate links in the published dataset, compromised quality of the dataset and global impact of the LOD dataset in terms of replicability of the overall process.

In order to solve the above-mentioned lacunas, the paper proposes a novel methodology called *LinkED (Linked Enterprise Data)* that contributes to the existing literature in two ways. First, the methodology streamlines the process of publishing enterprise data as LOD through five stages comprising cleaning, triplification, interlinking, storage and visualization. Second, the methodology addresses the latest LOD publication challenges, namely:

a) inadequate links,

b) inconsistencies in the quality of the dataset and

c) replicability of the overall process, highlighted by Hogan *et al.* [24], through the five stages.

Further, the paper demonstrates the methodology in a university setting by publishing digital repository data as LOD and evaluates the same based on two de facto semantic standards: Five-Star model, proposed by Tim Berners-Lee [25] and data quality metrics, proposed by Zaveri *et al.* [26].

The research contribution can be utilized in the following ways:

a) the proposed methodology serves as a template that can easily be customized for any future scenarios in LOD publication process;

b) although the publication process is demonstrated for education cluster (domain), the process can be replicated across other subject clusters in LOD such as healthcare, life sciences, open government, gene technology etc., as identified by Niknia and Mirtaheri [27].

The structure of the rest of the paper is as follows. Section 2 provides the background of the research. Section 3 describes the LinkED methodology for publishing enterprise data as LOD. Section 4 demonstrates a case study and further discusses the research contribution of the LinkED methodology. Section 5 presents a comparison with the related work and, lastly, Section 6 concludes the paper enlisting some future enhancements.

2. Background

In this section, first, we briefly review the current state-of-the-art literature regarding the adoption of LOD in enterprises. Second, we provide a brief survey of open source tools commonly used in the LOD publication process.

2.1. LOD Adoption in Enterprises

In the following, we investigate the adoption of LOD prescriptions in enterprises wherein technology is used to support ongoing business activities. LOD adoption in enterprises can be categorized as follows.

2.1.1. Expert Search

This category involves searching the enterprise and the Web based on a keyword commonly referred to as a tag, hence the search can be called "tag based search". The search is beneficial to the enterprise users who expect all relevant data to be displayed matching the tag. Bianchini [10] has proposed expert search in the enterprise and the Web through LINKSMAN (Linked data supported Mashup Collaboration) approach wherein search is conducted both internally within the enterprise as well as externally on the Web. Aastrand *et al.* [11] proposed an enterprise tagging mechanism to enterprise content stored in Drupal content management system [28]. The paper has used DBpedia RDF as the primary data source to obtain medical domain related information.

2.1.2. Unified Information Space in an Enterprise

The adoption of LOD prescriptions in enterprises complements technologies such as Master Data Management (MDM), Service Oriented Architecture (SOA) and Big Data, in creating a unified information space in the enterprise [9], [12]. However, the LOD approach is advantageous in many ways compared to the other technologies. MDM involves significant altering of the existing data model of the enterprise and considers only structured data sources for data pooling. In contrast, the LOD approach does not require altering of the existing data model and can consider structured, semi-structured and unstructured data.

SOA empowers peer-to-peer integration of silo applications in the enterprise and is generally not used for creating new data. In comparison to SOA, LOD provides a comprehensive view of the data as a whole and can also be used to create new information [9]. Big Data is used in enterprises to process massive amounts of data from heterogeneous sources that demand complex technological approach. Transforming the heterogeneous sources into linked data helps in inferring new associations and creating new knowledge that eventually helps in making smarter decisions in enterprises [29]. Thus LOD delivers "smart data" rather than just big data.

2.1.3. Augmentation of Core Data

Liu *et al.* [13] proposed a Web mashup approach that takes a keyword as input data and augments the data by retrieving relevant data from three disparate applications of YouTube, eBay and Flickr. The application interfaces with RESTful Web services exposed by the three applications and displays the retrieved data in a Web mashup. Voigt *et al.* [14] have proposed an enterprise Wiki approach for augmenting existing data wherein the local Wiki serves as an information encyclopedia within the enterprise.

2.2. Open Source Tools

In the following, we review the functionalities of some open source tools that are used in the various stages of our methodology.

2.2.1. OpenRefine

OpenRefine [30] is a tool from Google that assists in exploring, cleaning, reconciling and transforming messy datasets within enterprises. The tool accepts CSV, XML, JSON, RDF triples, spreadsheet formats as input, performs data cleanup, transformation (normalizing and denormalizing functions) using Google Refine Expression Language (GREL) and exports the refined data in TSV, CSV, Excel and HTML table formats. The tool is useful in the cleaning stage of the methodology for removing data inconsistencies within the enterprise data.

2.2.2. D2R Server

D2R Server [31] is a tool that publishes relational database content in RDF form. The tool aids in the mapping between database entities and Web resources that are described using semantic ontologies. The tool facilitates URI allocation, RDF hyperlink navigation and URI dereferencing, wherein any Web agents, crawlers or browsers can retrieve RDF and XHTML representations of the underlying resources. The tool also provides a SPARQL endpoint [32] for querying the RDF data through SPARQL queries [33] and displays the results in various visualization formats.

The process of transforming the raw enterprise data into RDF format is called *triplification* and the D2R Server is useful in the triplification stage of the methodology for URI allocation, URI dereferencing and conversion of the enterprise data into the RDF format.

2.2.3. SILK

SILK [34] is a framework for link discovery that employs Silk Specification Language to define RDF links between two or more datasets. The framework includes a workbench that offers a graphical editor for managing data sources, performing transformations, identifying similarity metrics and subsequently generating links between the datasets. Furthermore, SILK framework uses a pre-matching algorithm to compare different RDF datasets and employs BM25 weighting scheme for the ranking of search results, which is calculated based on the similarity metrics. The tool is useful in the interlinking stage of the methodology for linking the enterprise data with other data sources on the Web.

2.2.4. Triple Stores

The *Triple Stores* [35] or RDF stores are purpose-built databases for the deployment and retrieval of RDF triples through SPARQL queries. Some open-source examples include Virtuoso Universal Server, Open RDF, W3C Triple stores, DIQA basic stores. The Virtuoso Universal Server hosts DBpedia application which exposes Wikipedia information as RDF data. The Open RDF and W3C triple stores are public servers which allow any organization or individuals to freely host their RDF data on their servers. The DIQA basic stores contain a semantic data Wiki application made public via a SPARQL endpoint. In the proposed research, the triple stores are used in the storage stage of the methodology for the deployment of the RDF triples.

2.2.5. Visualization Tools

Most triple stores also act as *visualization tools* that display SPARQL query results in various formats such as HTML, RDF/ XML, Turtle, N3, N-Triples and CSV, to name a few. Sgvizler is a JavaScript library that renders charts and graphs based on SPARQL queries [36]. Sgvizler extends the Google Chart API and supports 1-dimensional, 2-dimensional, multi-dimensional, temporal and hierarchical data types. The visualization tools are useful in the visualization stage of the methodology for displaying RDF content in various presentation formats.

3. LinkED Methodology

LinkED methodology is designed as a linear-sequential process comprising cleaning, triplifi-

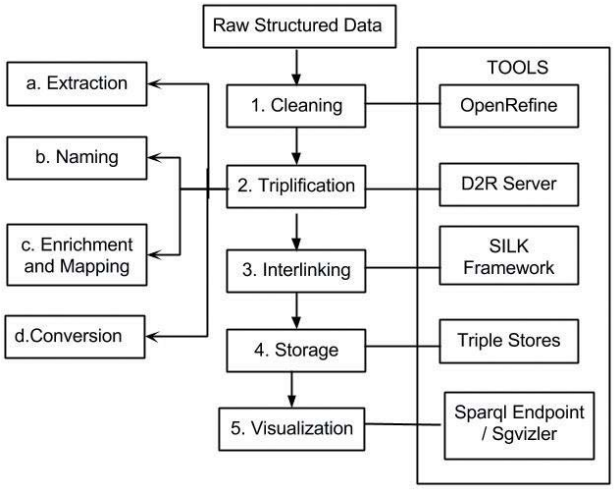

Figure 1. LinkED Methodology.

cation, interlinking, storage and visualization stages, which collectively publish linked enterprise data. The methodology starts with the identification of different data sources within the enterprise, whose data have the potential to be expressed as LOD.

Zaveri *et al.* [26] have observed serious lapses in the data quality of unstructured and semi-structured data as inputs to LOD and asserted that only structured data can be used as input. Hence, we consider only structured data to demonstrate the methodology. In the following, we describe different stages of the methodology, as outlined in Figure 1, and additionally list open source tools for implementing each stage of the methodology.

Stage 1: Cleaning. The identified enterprise data from different data sources are called "raw data" since they are unprocessed. The raw data needs to be first cleaned in order to make it coherent and ready for the next stages of the methodology since it may contain several inconsistencies and may lack the standard representation format. Since a majority of the data is generally observed to be of string, integer, or date types, an effective cleaning method is to enforce a consistent format for each data type. Some examples include:

a) mandating all date types to be in "dd/mm/yyyy" format, consistently across all the records;

b) mandating telephone numbers to contain an international prefix.

Overall, cleaning process ensures data standardization and consistency across various data sources.

Stage 2: Triplification. The transformation of raw data into RDF format is called triplification. This stage comprises four phases, namely: *extraction*, *naming*, *enrichment/mapping* and *conversion*, which collectively transform the raw data into a set of [subject, predicate, object] triples, as described in the following:

Phase 1: Extraction. The cleaned raw data is extracted from its data source into open, structured and non-proprietary formats, typically as CSV files or MySQL database. Certain amount of data transformation and a possible addition of metadata may be required prior to extraction. Since a *resource* forms the fundamental entity in the semantic world, the extraction phase is crucial in transforming the raw data into a suitable form, from which the resources can be identified.

Phase 2: Naming. This phase involves the identification and naming of unique resources within the extracted data. These resources shall form the subject and object components of the RDF triple later during the conversion phase.

Since a resource can represent a physical entity such as a document or a person, as well as abstract or conceptual entities, two kinds of URI naming standards can be used to represent each resource, namely, *slash (or 303) URI* format and *hash URI* format.

The difference between the two fragment identifiers (slash and hash) is related to a *303 redirect procedure*. The resources which are named with the slash identifier undergo double redirection when requested since the first redirection returns a "303 see other" status, which in turn redirects to a separate URI that provides a description of the requested resource. For example, the URI: http://eprints.manipal.edu/id/document/85361 is named in slash format, since dereferencing the HTTP URI yields information about the document with ID as 85361.

The use of a hash mark in URIs is to identify a portion of the resource. For example, the URI:

"http://eprints.manipal.edu/id/eprint/147312 authors" represents a portion of the resource "147312", i.e., a list of all the author resources for the eprint record.

Table 1. Key ontologies for enrichment.

Ontology Prefix	Namespace	Description
DC (Dublin Core) [41]	http://purl.org/dc/elements/1.1/	Describes generic metadata
SKOS (Simple Knowledge Organization System)	http://www.w3.org/2004/02/skos/core#	Describes knowledge organization systems
FOAF (Friend of a Friend) [38]	http://xmlns.com/foaf/0.1/	Describes persons, their activities and relations with other people
DOAP (Description of a Project)	http://usefulinc.com/ns/doap#	Describes a project profile
GN (Geonames)	http://www.geonames.org/ontology	Describes geospatial semantic information of the Word Wide Web
BIBO (Bibliographic Ontology) [42]	http://purl.org/ontology/bibo/	Describes bibliographic information
AIISO (Academic Institution Internal Structure Ontology) [39]	http://purl.org/vocab/aiiso/schema#	Describes the internal organizational structure of an academic institution
EP (EPrints) [40]	http://eprints.org/ontology/	Describes institutional repositories and scientific publications

Phase 3: Enrichment and Mapping. In this stage, the identified resource is enriched with well-known ontologies, which can either be *upper (top-level)* or *domain-based* in nature. Upper ontologies can be reused across a multitude of domains. Examples include DBpedia ontology [37], Friend of a Friend (FOAF) ontology [38], whose classes and properties are applicable across education, sports and business domains, to name a few. In contrast, domain ontologies are applicable only across a single domain. Some examples include Academic Institution Internal Structure Ontology (AIISO) [39] and EPrints ontology [40], which are applicable only in the education domain. A few of the key ontologies, along with their descriptions, are provided in Table 1.

Since we are dealing with structured data as input, presumably in CSV or relational database formats, we perform two types of enrichment viz. at the *class* level and at the *property* level. First, we ascertain the type of the entity, i.e. table/column within a relational database or file name/field within a CSV file. Next, we identify the potential ontology classes and/or properties with which the entity can be enriched.

The process of associating the entities with the identified ontology classes and properties is termed as "mapping". For example, let us consider a MYSQL database containing "author" table with two columns of "name" and "book title". Since the FOAF ontology contains classes and properties that describe people [38], we can enrich the author entity with relevant FOAF classes and properties. Subsequently, we can map the *author* entity with *foaf:Person* class, the author's name with *foaf:name* property and the book title with *bibo:title* property, since BIBO is a well-known ontology that describes bibliographic information [42].

Additionally, we can also describe data types for each mapping using XML data types such as xsd:integer, xsd:dateTime, xsd:string etc. In the previous example, the author name and the book title can be expressed as xsd:string types.

In order to better demonstrate the naming, enrichment and mapping activities, we explain a sample mapping file, created in Turtle format using the D2RQ Mapping Language [43] in Figure 2.

The Mapping.ttl file demonstrates a sample mapping for the database entities "person" (table entity) and "family_name" (column entity within the table). The database entities are enriched with the FOAF ontology and mapped as RDF terms using d2rq:ClassMap and d2rq:PropertyBridge constructs of the D2RQ mapping language. The person entity is identified as a class of type *foaf:Person*, having resources which follow the slash URI naming pattern:

```
# Mapping.ttl

# Table person
map:Person a d2rq:ClassMap;
d2rq:class foaf:Person; # rdf:Type
d2rq:uriPattern
"http://eprints.manipal.edu/id/person/
        @@person.personId@@"; .

# Column family_name
map:Family_name
a d2rq:PropertyBridge;
d2rq:belongsToClassMap map:Person;
d2rq:property foaf:familyName;
d2rq:column "person.family_name";
d2rq:datatype xsd:string; .
```

Figure 2. Sample mapping file.

"http://eprints.manipal.edu/id/person/@@person.personId@@". Furthermore, the URI "http://eprints.manipal.edu/id/person/" represents the base URI, in which resources are generated from values within the personID column.

The entity family_name is related to the Person class and identified as a property of type *foaf:familyName*. The property contains literal values of type "xsd:string", that comes from the database column named "person.family_name".

Overall, the outcome of the enrichment and mapping stages is a file (such as the Mapping.ttl shown in the example) which contains all the enrichment/mapping details required for the conversion of the extracted data into the RDF format.

Phase 4: Conversion. This stage involves the actual transformation of the raw data into RDF dataset based on a serialization format. Popular RDF serialization formats include RDF/XML, Turtle, N-Triples and N3. All formats support SPARQL query language for retrieving and manipulating the RDF data.

Stage 3: Interlinking. This stage facilitates *enterprise-Web* interoperability by establishing semantic links between the source dataset (enterprise data) with other potential target datasets on the World Wide Web. As the first step in interlinking, due diligence is performed wherein potential target data sources are identified, as shown in Figure 3. Next, the activities of *link discovery* and *link generation* are performed. While the link discovery activity aids in discovering links between the source and target datasets, link generation activity generates the links in RDF format using *rdfs:seeAlso* or *owl:sameAs* predicates.

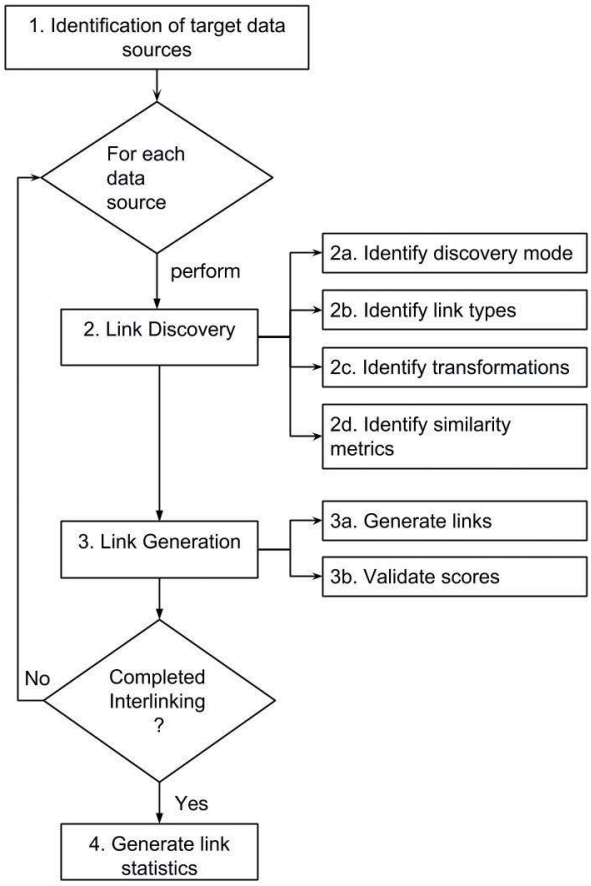

Figure 3. Interlinking stage of LinkED Methodology.

Link discovery activity can be performed in manual, semi-automated or fully-automated modes. Manually discovering links is tedious and error-prone since RDF deals with thousands/millions of triples. The semi-automated mode is reliable and preferred since most link discovery tools available in the market require some amount of manual intervention. Currently, fully-automated link discovery engines are unavailable.

After finalizing a discovery mode, we identify the types which can be matched between the source and target datasets, and the required transformations, if any. The types include classes and properties in the source and target ontologies. For example, *foaf:name* property belonging to *foaf:Person* class within the source dataset can be identified for linking with the

foaf:name property in *foaf:Agent* class within the target dataset. Subsequently, we can apply transformations on the types, either the source and/or target datasets, wherein the data can be converted into suitable formats that aid the linking activity. Examples of transformations include: converting the data into lowercase, removing unwanted values such as blank or special characters, applying regular expression rules etc. The last sub-activity in link discovery is the identification of similarity metrics. Typically, "string", "integer" and "date" are the frequent data types that are encountered in source and target datasets. For the "integer" and "date" types, a direct comparison can be performed. However, for the string types, character-based comparator algorithms such as Levenshtein, Smith-Waterman, Jaro, Jaro-Winkler etc., can be applied [44], [45].

Character-based comparator algorithms compare strings at the character level [44]. They are best suited for handling typographical errors such as:

a) additional spacing between words,
b) punctuations,
c) rearrangement of words,
d) small character changes, and
e) presence of insignificant words/characters [46].

Levenshtein and Smith-Waterman are "edit-distance" based algorithms [44]. In order to compare the two input strings (from the source and target datasets), the edit-distance based algorithms first compute a *minimum edit distance*, which is based on the number of operations required to transform the source string into the target string. The operations include insertion, deletion and substitution of characters. Finally, based on the minimum edit distance, a score is computed that indicates the similarity between the two strings. Scores vary in the range [0…1], with values close to +1 indicating that the strings are similar. Typically, the edit-distance based algorithms are effective in detecting small character changes and rearrangement of words [46].

The Jaro and Jaro-Winkler algorithms are character-based distance measures that were specifically developed for name comparison in the U.S. Census [44]. These algorithms consider character transpositions, which are based on the number and order of common characters between the source and target strings [44]. Typically, these algorithms are best suited for handling additional spacing between words, punctuations and presence of insignificant words/characters [46].

Once the comparator algorithm is selected, threshold levels can be set for each of these algorithms, above which the string matching is considered valid. After choosing an appropriate similarity metric, links and matching scores are generated, which satisfy the threshold levels.

The activities of link discovery and link generation are performed sequentially for each data source. The last activity within the interlinking stage is the generation of overall link statistics which showcase the total number of links generated between the source and target data sources.

Stage 4: Storage. The process of storing the interlinked dataset in RDF format is termed as "storage". Typically, a "triple store", which represents a specialized database, is used for storing the RDF dataset [35]. Retrieval of RDF triples from the storage 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.

Stage 5: Visualization. The presentation of RDF data in HTML, JSON, CSV or XML formats is termed as "visualization". Since RDF dataset is primarily meant for machine-readability and not for human-readability, RDF data presented in HTML, JSON or XML formats may not make much sense to the end-user. Hence, visualization can additionally include the showcasing of RDF data intuitively via charts, graphs and maps, thus facilitating end-users towards experimentation and knowledge discovery of the dataset. Additionally, dataset statistics can also be displayed.

Each stage of the LinkED methodology can be implemented using open source tools described in Background section, whose correlation with the methodology stages is shown in Figure 1.

4. Case Study, Evaluation and Discussion

This section describes a case study in education domain that demonstrates all stages of the LinkED methodology. The case study is evaluated for adherence to Five-Star model and data quality metrics. The section also discusses the overall contribution of the LinkED methodology to the LOD literature.

4.1. Case Study

A case study was conducted in the education domain by considering structured data from *eprints* [47] enterprise software for publication as LOD. In any educational setting, publication information is considered vital in showcasing the strength of the university. Hence, we chose to demonstrate our methodology using "eprints", which is a leading digital repository used by 12,000+ institutions across the world for storing publication information [48]. Manipal University uses eprints for storing research publications such as journals, conferences, books, theses etc. [47]. The goal of the case study was to publish the eprints data as LOD, by following the LinkED methodology steps. In the following, we describe details of the case study.

In the *cleaning stage*, the eprints data, residing in MySQL database, was recognized as "raw structured data". The data contained inconsistencies such as:

a) non-standardized representation of date format,

b) naming inconsistencies, and

c) missing subject fields across eprint records.

We performed data cleaning activity, which resolved the above-mentioned inconsistencies. We restricted the date format as per two possible formats: "dd/mm/yyyy" or "yyyy", based on the available date information. With regard to naming inconsistencies, we noticed that the values in the column "given_name" were not consistent across different eprint records. Since only "given_name" and "family_name" fields existed in the database, and there was no column to store the "middle name" of the author, the inconsistency existed.

The naming inconsistencies were resolved by intimating the librarians, who maintain eprints application. They proposed a standard format for naming conventions and also rectified the majority of naming errors. With respect to missing subject fields, totally, there were 9 eprint records with missing subject field. The list was given to the librarians, who updated the missing subject fields.

In the *triplification stage*, the process of extraction was not required, since the data was already in open, structured and non-proprietary format.

In the naming phase, resources were identified and, subsequently, URI patterns were created, as shown in Table 2. The slash naming format was used for naming the "eprint", "subject" and "person" resources, while the hash naming format was used for representing the list of authors for an eprint record and the list of subjects for an eprint record.

The enrichment activity ensured that the raw eprints data was enriched with well-known ontologies such as EP, DCT, FOAF, SKOS etc.

Next, a mapping file was created in Turtle format which mapped the database entities (tables and columns) to RDF resources. Specifically, the tables within the database were mapped to relevant classes of different ontologies, while the columns were mapped to relevant properties. As shown in Table 2, the "Eprint" database entity was mapped to *ep:Eprint* and *bibo:AcademicArticle* classes; the "Subject" entity to *skos:Concept* class and the "Person" entity to *foaf:Person* class. Every eprint publication record (journal/conference/book) contains "abstract" and "title" information. As shown in Table 2, the "Abstract" column was mapped with *bibo:abstract* property and "Title" column with "dct:title" property respectively, wherein both BIBO and DCT are upper ontologies.

Additionally, multiple joins and hash functions were performed on the columns, whose details are listed in Table 2. The complete mapping file, in Turtle format, is made available at the URL: "https://tinyurl.com/epMappingFile".

Further, in the conversion phase, the mapping file was used as input to create an RDF file in N-Triples format using the D2R server. We chose N-Triples format over other serialization formats since N-Triples is a subset of Turtle having no parsing directives, thus making the process of serialization the fastest among its

Table 2. Key mapping details.

Resource name: eprint. URI pattern: http://eprints.manipal.edu/id/eprint/ RDF Type: ep:Eprint, bibo:AcademicArticle				
Database entity	**Database entity type**	**Ontology entity**	**Ontology entity type**	**Description**
Eprint	Table	ep: Eprint; bibo:AcademicArticle	Class	Publication record
Eprintid	Column	ep:eprintid	Property	Unique Id for the eprint
Abstract	Column	bibo:abstract	Property	Abstract of the publication
Title	Column	dct:title	Property	Title of the publication
Creators_name_given, Creators_name_family	Columns	dct:creator	Property	First name and family name of the authors who published the eprint record
Subject	Column	dct:subject	Property	Subject type of the publication
Date	Column	dct:date	Property	Date of publication
Resource name: **subject**. URI pattern : http://eprints.manipal.edu/id/subject/ RDF Type: skos:Concept				
Database entity	**Database entity type**	**Ontology entity**	**Ontology entity type**	**Description**
Subject	Table	skos:Concept	Class	Subject record
Subjectid	Column	ep:subjected	Property	Unique ID for the subject
Name	Column	skos:prefLabel	Property	Name of the subject
Join: eprint.subject = subject.subjectid				
Resource name: **person**. URI pattern : http://eprints.manipal.edu/id/person/ RDF Type: foaf:Person				
Database entity	**Database entity type**	**Ontology entity**	**Ontology entity type**	**Description**
Person	Table	foaf:Person	Class	Author record
Creators_name_given	Column	foaf:givenName	Property	Given name of the author
Creators_name_family	Column	foaf:familyName	Property	Family name of the author

contemporaries. The speed of serialization is an important factor during the interlinking phase wherein the eprints data is compared with millions of other triples from different sources in order to recognize potential linking opportunities.

As part of the *interlinking stage*, we used semi-automated techniques for link discovery between the eprints dataset and other datasets on the Web, primarily DBLP [49] and DBpedia [50] datasets. The DBLP dataset contains computer science bibliography information, while the DBpedia dataset represents a linked-data extraction from Wikipedia. The eprints dataset was linked with the DBLP dataset on author names and the DBpedia dataset on subject names.

SILK workbench was employed to establish RDF links between eprints' *foaf:Person* class and DBLPs' *foaf:Agent* class, wherein both classes contained author information. We used lowercase transformation to match the person and author names. Furthermore, the *Jaro-Winkler* metric was chosen to compute the similar-

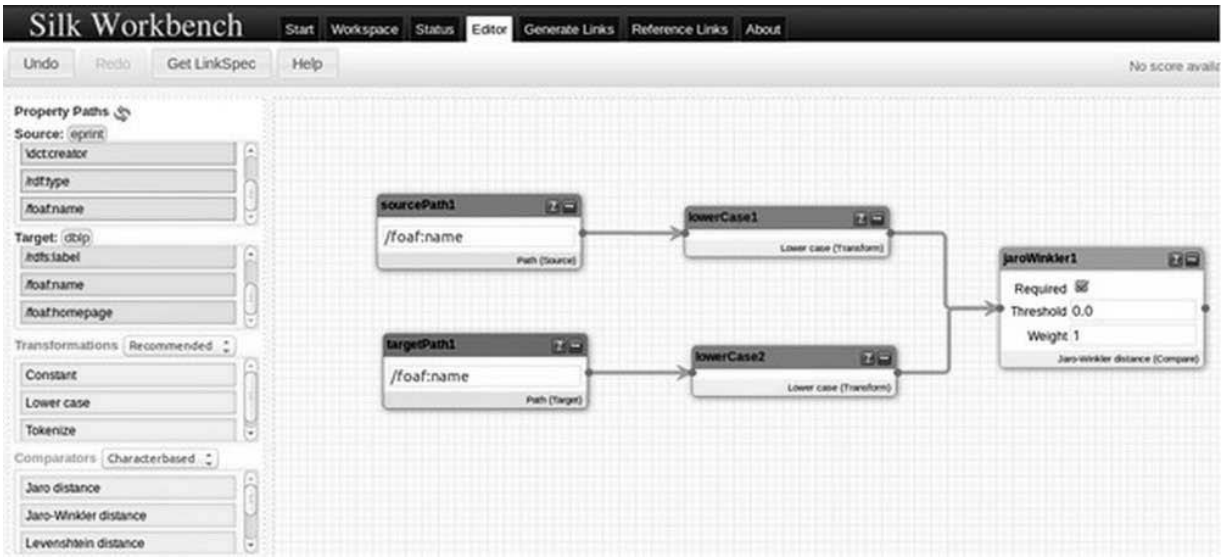

Figure 4. Transformation and Similarity matching.

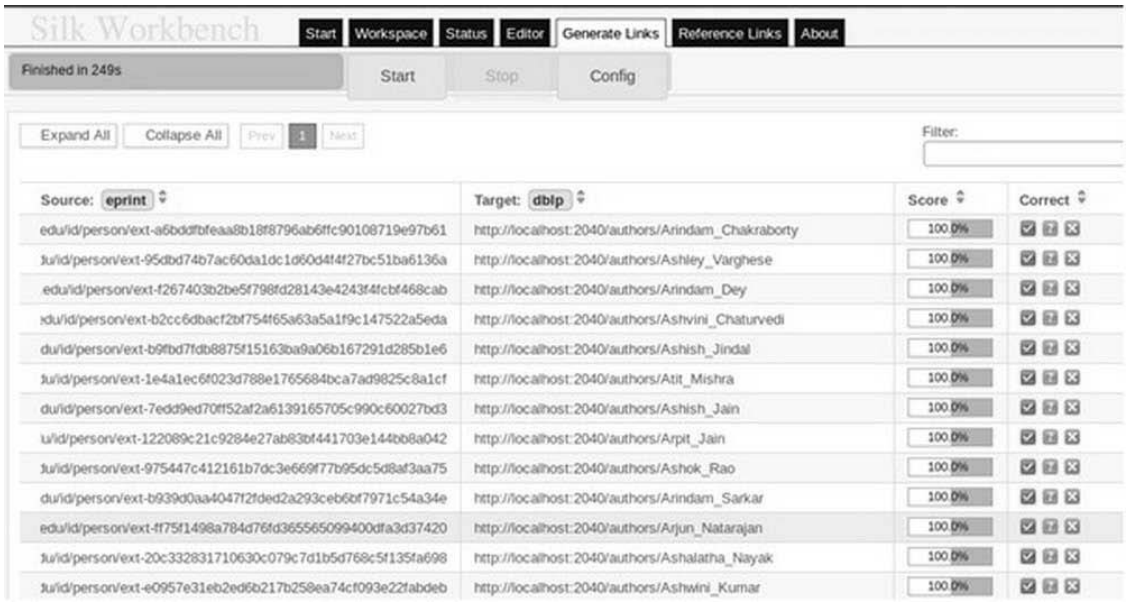

Figure 5. A sample result of link generation.

ities between the person name and author name strings, as indicated in Figure 4.

The rationale for choosing *Jaro-Winkler* as the string comparator algorithm is because the metric considers *character transpositions* between the source and target strings, hence best suited for short strings such as person names [45]. For example, let us consider the name of the author, i.e., "Shreyas Rao" as the source string in the eprints dataset. The author name in the target dataset could be "Shreyas S Rao", or "Shreyas, Rao" or "ShreyasRao". The variants indicate addition of new characters ("S" in Shreyas S Rao), punctuation (Shreyas, Rao) or spacing (ShreyasRao). The Jaro-Winkler algorithm provides scores of 0.97, 0.98 and 0.98 respectively for the three variants which are the highest similarity scores among all other string comparator algorithms [46].

As a consequence of the interlinking activity, we found *637 links* between eprints and DBLP resources. Ultimately, the eprints and DBLP resources were linked using *rdfs:seeAlso* predicate. A sample result of link generation is shown in Figure 5.

A similar linking exercise was carried out between eprints and the DBpedia datasets. Out of 260 subjects in the eprints dataset, we obtained *117 links* with DBpedia dataset. The "subject" resource of eprints was linked with entities of type "Thing" or "Concept" within DBpedia using *owl:sameAs* predicate.

Next, in the *storage stage*, the eprints dataset was loaded onto the Openlink Virtuoso Universal Server and made accessible via a SPARQL endpoint locally within the Manipal University Intranet. The eprints dataset is publicly available at the URL "http://tinyurl.com/epDataSet". The overall statistics for the eprints deployment is presented in Table 3.

Table 3. Overall statistics of the eprints dataset.

Description	Count
Total number of RDF triples	406565
Total number of entities	30406
Total number of classes	11
Total number of properties	65
Total number of distinct subjects	30578
Total number of distinct objects	126480
Total number of links with DBLP	637
Total number of links with DBpedia	117

The eprints dataset, deployed on Virtuoso Universal Server, supports visualization formats such as HTML, XML, JSON, JavaScript, Turtle, RDF/ XML, N-Triples, CSV, Spreadsheet and TSV for displaying SPARQL results. In addition to the *visualization* provided by the Virtuoso Universal Server, we have developed a C# and JavaScript based application that assists end users in knowledge discovery of the eprints dataset intuitively via charts and graphs. In order to help the end-user with SPARQL queries, we have provided sample queries categorized as general queries, researcher-based queries, enterprise collaboration based queries and statistics on eprints dataset. Sample charts generated for researcher-based queries and enterprise collaboration based queries are shown in Figures 6 and 7. Figure 6 depicts a pie chart showing top 5 collaborators of a sample researcher.

Figure 7 depicts a line chart showcasing year-wise collaborations of Computer Science and Engineering department with other departments in Manipal University. The SPARQL query considers co-authorship in publishing journal papers, conference papers and book sections as valid collaborations.

4.2. Evaluation

A twofold evaluation is performed for the case study. First, we compare the adherence of the case study with the Five-Star Model [25], which is the *de facto* standard for publishing LOD on the Web. Second, we evaluate the quality of the eprints dataset based on the metrics proposed by Zaveri *et al.* [26].

4.2.1. Adherence with the Five-Star Model

The Five-Star Model [25] represents a deployment scheme for open data on the Web. According to the model, 1-star denotes making the data available on the Web, in any format (such as pdf), with an open license. 2-star denotes representing the data in a structured format (such as Excel); 3-star denotes representing the data in a non-proprietary open format (such as CSV); 4-star requires assigning URIs to data items and, lastly, 5-star requires the data to be linked with other relevant data on the Web, based on the context. A five-star compliant data facilitates the discovery of more interlinked information on the Web and inference of new knowledge from existing facts [51].

In our case study, the raw data chosen from eprints for triplification is open (eprints application), structured (MySQL database), and in non-proprietary format (MySQL). Thus the data adheres to 1-star, 2-star and 3-star compliances. In the conversion stage, the mapped data is converted into N-Triples format, which satisfies the 4-star compliance having unique URIs denoting entities within the eprints dataset. Lastly, in the interlinking stage, the URIs are linked with appropriate ontologies on the Web using *owl:sameAs* and *rdfs:seeAlso* predicates, thereby fusing the source triples with relevant triples in the Web and satisfying the 5-star compliance. Also, we have used the OpenLink Virtuoso Universal Server for the deployment of eprints dataset, which is Five-Star LOD compliant [52].

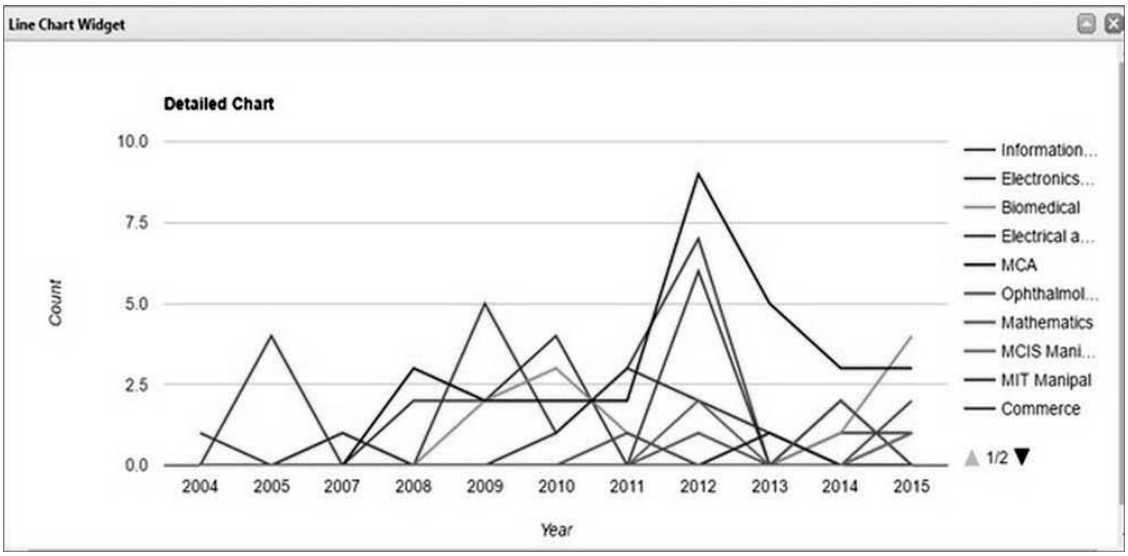

Figure 7. Line chart depicting year-wise collaborations of Computer Science and Engineering department with other departments in Manipal University.

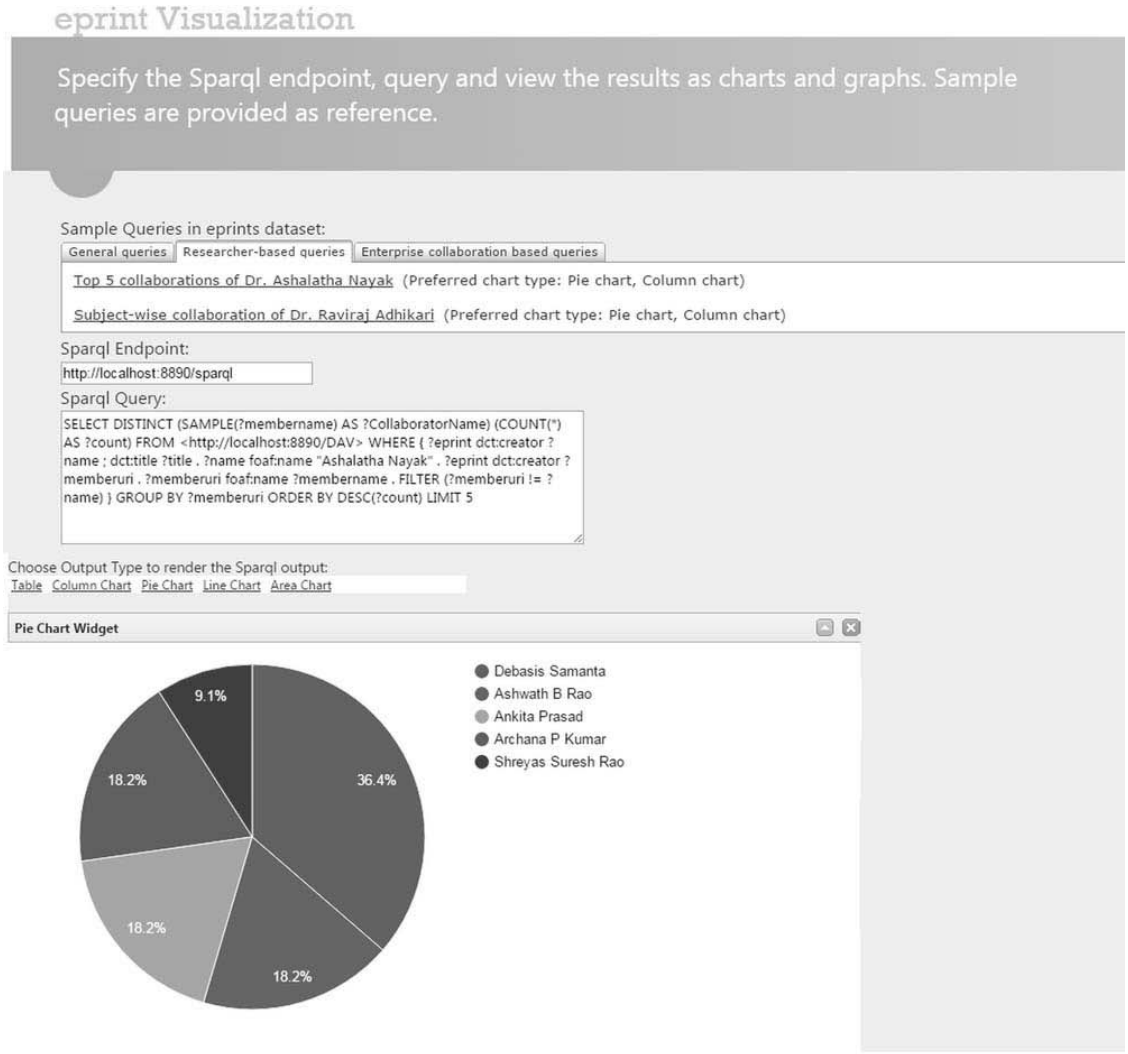

Figure 6. Screenshot of eprints application showing a pie chart of top 5 collaborators of a researcher.

4.2.2. Data Quality

Assessing the quality of the linked dataset is of paramount importance in judging its utility and reusability [26]. Zaveri *et al.* [26] provided a list of quality dimensions and associated metrics, which collectively evaluate the quality of the published dataset. We have evaluated the eprints dataset according to the dimensions of availability, interlinking, interoperability, versatility and licensing, as shown in Table 4.

Table 4. Quality evaluation for the eprints dataset.

Metric	Description of the metric	Adherence confirmation
Availability	Accessibility of the RDF dumps	RDF dump in N-Triples format is accessible at URL "http://tinyurl.com/epDataSet" in Microsoft OneDrive
	Dereferenceability of the URI	RDF data is returned on accessing any URI
Interlinking	Existence of links to external data providers	eprints dataset is linked with DBLP and DBpedia datasources using rdfs:seeAlso and owl:sameAs predicates
Interoperability	Re-use of existing terms from relevant ontologies in the specific domain	eprints dataset reuses terms from http://eprints.org/ontology/ ontology which represents the domain of Institutional publications
	Re-use of existing ontologies	eprints dataset reuses existing terms from well-known ontologies such as dcterms, foaf, skos, bibo, geonames etc.
Versatility	Provision of the data in different serialization formats	eprints dataset is available as RDF dump in the following formats: N-Triples, N3, Turtle and RDF/XML
Licensing	Indication of an open license for the software	eprints ontology and software are available under an open license [48]

uses the D2R Server during the interlinking stage and Virtuoso Universal Server during the storage stage, both of which provide SPARQL endpoints and help navigate the IRIs through SPARQL queries.

The second issue concerns the quality of the datasets wherein inconsistencies are observed during the naming of entities and IRI dereferencing. In the LinkED methodology, naming, enrichment/mapping and conversion phases within the triplification stage mandate consistent naming of entities and IRI dereferencing.

The third issue concerns the impact of the LOD publishing process, wherein multiple replications of the process must be possible. The LinkED methodology, demonstrated via the "eprints" case study, is novel and the process can be replicated across the 12,000+ institutions that use eprints software. Thus, the LinkED methodology addresses the latest LOD publication challenges.

4.3. Discussion

In this section, we discuss how the LinkED methodology addresses unresolved issues in LOD publishing, provides a generic and streamlined process for LOD publishing, and handles semi-structured and unstructured data formats as input.

4.3.1. Addressing of Unresolved Issues in LOD Publication

The LinkED methodology addresses the unresolved and recurrent issues in LOD publication. The first issue deals with inadequate links in the published datasets. Out of 31 SPARQL endpoints considered by Hogan *et al.* [24], 13 endpoints (i.e. 41.93%) had inadequate link problem, wherein the authors had difficulty in accessing the IRIs. LinkED methodology

4.3.2. Generic and Streamlined Process for LOD Publishing

In the publication process of enterprise data as LOD, disparate enterprises deem different stages as critical. It is observed that government open data is often prone to errors [15], [20],

[21] and *cleaning* stage is very critical. Also, government data is spread over multiple systems and *extraction* from different data sources forms an important step in the LOD publication process. However, in pharmaceutical research [17] and clinical trials [16], cleaning is not a problem, but *interlinking* with other resources on the Web is a bigger challenge. In transforming meteorological data into linked data [23], *naming* and *enrichment/mapping* stages are crucial and often determine the success of the LOD effort. In contrast, in Destination Management Organizations (DMO) [22], *storage* of millions of customer records and *visualization* of the respective LOD datasets forms the crux in the publishing process. LinkED methodology incorporates all these necessities and criticalities from various domains, and is thus generic, comprehensive and applicable across a plethora of enterprises such as healthcare, tourism, government, pharmaceutical, banking and automobile, to name a few.

4.3.3. Semi-structured and Unstructured Data as Inputs to LinkED Methodology

In the LinkED methodology, the scope of the enterprise data is limited to "text", which can appear in unstructured (documents such as PDF/Word etc.), semi-structured (XML/JSON etc.) or structured (relational database/CSV etc.) formats. Since the LinkED methodology described in the paper considers structured data as input, we discuss other possibilities in the following.

There are two approaches in dealing with semi-structured data as input to the methodology: (a) convert the data into structured format using open-source converters [53], [54], [55], and then provide the converted data as input to the methodology or (b) convert the data directly into RDF triples format using the xCurator framework [56].

In the first approach, converter tools such as SQLizer [53], JSON to CSV converter [54] and Advanced XML converter [55] can be used to convert the input data in XML/JSON formats to corresponding CSV, Excel or MySQL (script) formats. The converted data then undergoes the cleaning, triplification, interlinking, storage and visualization stages of the LinkED methodology.

In the second approach, the semi-structured data is directly converted into RDF triple format using the xCurator framework [56]. The "entity type extractor" component of the framework [56] can be used for identification and naming of the resources, their classes, and properties, while the “transformer” component [56] can be used for converting the data into RDF triples format. Since the transformation process using the xCurator framework subsumes the "triplification" stage of the methodology, one can directly proceed with the interlinking, storage and visualization stages of the methodology.

Compared to the semi-structured data, the use of unstructured data as input to the methodology is more complex and challenging [57], [58]. Augenstein *et al.* [57] assert that the creation of an RDF representation from unstructured sources, particularly textual data, is a challenging task and has not been satisfactorily solved in literature. Hence, suggesting a satisfactory approach in dealing with unstructured data as input to the LinkED methodology is beyond the scope of this paper.

However, the "Text Tagging and Annotation" approach [59] and the "LODifier" approach [57], both involving the use of natural language processing techniques for unstructured to structured/ linked data conversions respectively, can be explored as future work.

5. Related Work

The LinkEd methodology is comparable to the work proposed by Garcia *et al.* [22] for the publication of LOD in the tourism industry. Correlating with the LinkED methodology, the "pre-processing" stage corresponds with our cleaning stage, the "configuration" with the mapping stage and "publication" with the storage stage. In addition, LinkED methodology defines an enrichment stage wherein ontologies are divided into upper and domain-based categories. This division is essential to enforce the use of upper ontologies during the publication process, in addition to the domain specific ontologies. Furthermore, our methodology provides an in-depth coverage of the interlinking stage subdividing into link discovery and link generation activities. Also, our work adds conversion and visualization stages, which are not present in Garcia *et al.*'s work.

Malli *et al.* [21] proposed a methodology for publishing linked government data through a "self-service" approach, wherein the data consumer can generate linked government data themselves without the intervention of the government. We argue that LinkED methodology is also a "self-service" approach wherein the end-users can execute different stages of the methodology without requiring much technical assistance since our methodology adheres to the five requirements enlisted by Malli *et al.* [21]. First, end-users have full control over the publication process, since every stage is designed to be manual or semi-automatic in nature, requiring human intervention. Second, all the tools prescribed in the methodology have a GUI and are easy-to-use for non-expert end-users, barring the Sgvizler JavaScript library [36] which needs prior knowledge of JavaScript. Third, the methodology is reproducible, as substantiated in the discussion section. Fourth, the methodology is flexible and decentralized, since all the tools used in implementing different stages of the methodology are independent of each other, thus provides a decentralized setting. Lastly, the results of the published RDF dataset can be shared via visualization in a multitude of formats, enabling any human or system (Web agent, crawler, browser) to query and reuse the data.

Currently, the scope of the LinkED methodology is limited to the publication phase within the overall LOD lifecycle [60]. However, the LOD lifecycle consists of other phases such as evolution, repair, manual revising (authoring) and quality analysis, which needs to be investigated further.

6. Conclusion

Advancement in semantic technologies over the past decade has fostered researchers to re-examine the enterprise-Web interoperability issue. Publishing enterprise data as linked open data aims to solve the interoperability issue by interlinking the enterprise data with the Web using ontologies. The paper proposes a methodology called LinkED that streamlines the LOD publication process and solves the latest LOD publication challenges. Furthermore, the LinkED methodology is demonstrated via an educational case study and evaluated based on the Five-Star model and data quality metrics. Overall, we believe that this research contributes to semantic Web literature in general and to LOD practice in specific, through the LinkED methodology.

The future work includes:

- Extending the LinkED methodology to include supervised machine learning of ontologies as part of the enrichment stage
- Evaluating the eprints dataset based on performance dimension
- Hosting the C# application on Cloud platform, enabling researchers to avail the visualization features of the eprints dataset
- Exploring the possibility of using unstructured data as input to the LinkED methodology

References

[1] G. Booch, "Enterprise Architecture and Technical Architecture", *Journal of IEEE Software*, vol. 27, no. 2, pp. 96, 2010. http://dx.doi.org/10.1109/MS.2010.42

[2] A. Holzinger *et al.*, "Combining HCI, Natural Language Processing, and Knowledge Discovery – Potential of IBM Content Analytics as an Assistive Technology in the Biomedical Field", in *Proceedings of the Human-Computer Interaction and Knowledge Discovery in Complex, Unstructured, Big Data*, Maribor, 2013, pp. 13–24. http://dx.doi.org/10.1007/978-3-642-39146-0_2

[3] L. Tang *et al.*, "Modeling Enterprise Service-Oriented Architectural Styles", *Journal of Service Oriented Computing and Applications*, vol. 4, no. 2, pp. 81–107, 2010. http://dx.doi.org/10.1007/s11761-010-0059-2

[4] M. Psiuk *et al.*, "Enterprise Service Bus Monitoring Framework for SOA Systems", *IEEE Transactions on Services Computing*, vol. 5, no. 3, pp. 450–466, 2012. http://dx.doi.org/10.1109/TSC.2011.32

[5] R. T. Fielding and R. N. Taylor, "Principled Design of the Modern Web Architecture", *ACM Transactions on Internet Technology*, vol. 2, no. 2, pp. 115–150, 2002. http://dx.doi.org/10.1145/514183.514185

[6] C. Su and C. Chiang, "Enabling Successful Collaboration 2.0: A REST-Based Web Service and Web 2.0 Technology Oriented Information Platform for Collaborative Product Development", *Journal of Computers in Industry*, vol. 63, no. 9, pp. 948–959, 2012. http://dx.doi.org/10.1016/j.compind.2012.08.018

[7] G. Antoniou and F. van Harmelen, "A Semantic Web Primer", MIT Press second edition, 2008.

[8] C. Bizer *et al.*, "Linked Data – The Story So Far", *International Journal on Semantic Web and Information Systems*, vol. 5, no. 3, pp. 1–22, 2009. http://dx.doi.org/10.4018/jswis.2009081901

[9] D. Wood (ed.), "Linking Enterprise Data", Springer Science & Business Media, 2010. http://dx.doi.org/10.1007/978-1-4419-7665-9

[10] D. Bianchini *et al.*, "A Linked Data Perspective for Collaboration in Mashup Development", in *Proceedings of the 24th International Workshop on Database and Expert Systems Applications*, Los Alamitos, 2013, pp. 128–132. http://dx.doi.org/10.1109/DEXA.2013.20

[11] G. Aastrand *et al.*, "Using Linked Open Data to bootstrap corporate Knowledge Management in the OrganiK project", in *Proceedings of the 6th International Conference on Semantic Systems*, Graz, 2010, Article no. 18. http://dx.doi.org/10.1145/1839707.1839730

[12] Antidot Technologies (2008). Linked Enterprise Data:Principles, uses and benefits [Online]. Available: http://www.antidot.net/wp-content/uploads/2012/11/LinkedEnterpriseData-WP-en-v2.2.pdf

[13] X. Liu *et al.*, "iMashup: a Mashup-Based Framework for Service Composition", *Journal of Science China Information Sciences*, vol. 57, no. 1, pp. 1–20, 2014. http://dx.doi.org/10.1007/s11432-013-4782-0

[14] S. Voigt *et al.*, "ICKEwiki: Requirements and Concepts for an Enterprise Wiki for SMEs", in *Proceedings of the 7th International Symposium on Wikis and Open Collaboration*, Mountain View, 2011, pp. 144–153. http://dx.doi.org/10.1145/2038558.2038582

[15] M. Kaschesky and L. Selmi, "Fusepool R5 Linked Data Framework: Concepts, Methodologies and Tools for Linked Data", in *Proceedings of the 14th Annual International Conference on Digital Government Research*, Quebec, 2013, pp. 156–165. http://dx.doi.org/10.1145/2479724.2479748

[16] O. Hassanzadeh *et al.*, "LinkedCT: A Linked Data Space for Clinical Trials", arXiv:0908.0567v1, 2009.

[17] M. Samwald *et al.*, "Linked Open Drug Data for Pharmaceutical Research and Development", *Journal of Cheminformatics*, vol. 3, no. 1, pp. 19, 2011. http://dx.doi.org/10.1186/1758-2946-3-19

[18] P. Groth and Y. Gil, "Linked Data for Network Science", in *Proceedings of the First International Conference on Linked Science LISC*, Aachen, 2011, pp. 1–12.

[19] I. Petrou *et al.*, "Towards a Methodology for Publishing Linked Open Statistical Data", J*ournal of Democracy and Open Government*, vol. 6, no. 1, pp. 97–105, 2014.

[20] B. Villazón-Terrazas *et al.*, "Methodological Guidelines for Publishing Government Linked Data", in D. Wood (Ed), Linking Government Data, Springer, pp. 27–49, 2011. http://dx.doi.org/10.1007/978-1-4614-1767-5_2

[21] F. Malli *et al.*, "A Publishing Pipeline for Linked Government Data", in E. Simperl *et al.* (Eds), The Semantic Web: Research and Applications, Springer, pp. 778–792, 2012. http://dx.doi.org/10.1007/978-3-642-30284-8_59

[22] A. García *et al.*, "Methodology for the Publication of Linked Open Data from Small and Medium Size DMOs", in *Proceedings of the Information and Communication Technologies in Tourism*, Lugano, 2015, pp. 183–195. http://dx.doi.org/10.1007/978-3-319-14343-9_14

[23] G. Atemezing *et al.*, "Transforming Meteorological Data into Linked Data", Semantic Web Journal, vol. 4, no. 3, pp. 285–290, 2013. http://dx.doi.org/10.3233/SW-120089

[24] A. Hogan *et al.*, "Linked Dataset Description Papers at the Semantic Web Journal: A Critical Assessment", *Semantic Web Journal*, vol. 7, no. 2, pp. 105–116, 2016. http://dx.doi.org/10.3233/SW-160216

[25] Five Star Open Data. [Online] Available: http://5stardata.info/en/

[26] A. Zaveri *et al.*, "Quality Assessment for Linked Data: A Survey", *Semantic Web Journal*, vol. 7, no. 1, pp. 63–93, 2015. http://dx.doi.org/10.3233/SW-150175

[27] M. Niknia and S. L. Mirtaheri, "Mapping a Decade of Linked Data Progress Through Co-Word Analysis", *Webology*, vol. 12, no. 2, Article 141, 2015.

[28] Drupal. [Online] Available: https://www.drupal.org/8

[29] S. Consoli *et al.*, "Producing Linked Data for Smart Cities: The Case of Catania", *Big Data Research*, vol. 7, pp. 1–15, 2017. https://doi.org/10.1016/j.bdr.2016.10.001

[30] R. Verborgh and M. De Wilde, "Using OpenRefine", Packt Publishing, 2013.

[31] C. Bizer and R. Cyganiak, "D2R Server – Publishing Relational Databases on the Semantic Web", in *Proceedings of the 5th International Semantic Web Conference (ISWC)*, Springer LNCS 4273, 2006.

[32] D2R Server: Accessing databases with SPARQL and as Linked Data. [Online] Available: http://d2rq.org/d2r-server#features

[33] SPARQL Query Language for RDF. [Online] Available: https://www.w3.org/TR/rdf-sparql-query/

[34] J. Volz *et al.*, "Silk – a Link Discovery Framework for the Web of Data", in *Proceedings of the 2nd Linked Data on the Web Workshop*, 2009.

[35] List of Triple Stores. [Online] Available: http://en.wikipedia.org/wiki/Triplestore

[36] M. G. Skjaeveland, "Sgvizler: A JavaScript Wrapper for Easy Visualization of SPARQL Result Sets", in *Proceedings of the 9th Extended Semantic Web Conference (ESWC)*, pp. 361–365, 2012. http://dx.doi.org/10.1007/978-3-662-46641-4_27

[37] Ontology Classes. [Online] Available: http://mappings.dbpedia.org/server/ontology/classes/

[38] FOAF Vocabulary Specification 0.99. [Online] Available: http://xmlns.com/foaf/spec/

[39] Academic Institution Internal Structure Ontology (AIISO). [Online] Available: http://vocab.org/aiiso/

[40] EPrints ontology. [Online] Available: http://bartoc.org/en/node/17953/

[41] DCMI Metadata Terms. [Online]. Available: http://dublincore.org/documents/dcmi-terms/

[42] Bibliographic Ontology Specification. Available: http://bibliontology.com/

[43] The D2RQ Mapping Language. [Online] Available: http://d2rq.org/d2rq-language

[44] W. W. Cohen *et al.*, "A Comparison of String Distance Metrics for Name-Matching Tasks", in *Proceedings of the 2003 International Conference on Information Integration on the Web (IWEB'03)*, Athens, Georgia, 2003, pp. 73–78.

[45] SILK – Character Based Distance Measures. [Online] Available: https://app.assembla.com/wiki/show/silk/Comparison

[46] Test Similarity. [Online] Available: https://asecuritysite.com/forensics/simstring?word=loans%20and%20accounts%24loans%20accounts

[47] Manipal University Digital Repository. [Online] Available: http://eprints.manipal.edu/

[48] EPrints for Open Access. [Online] Available: http://www.eprints.org/uk/index.php/openaccess/

[49] DBLP RDF Dump. [Online] Available: http://dblp.l3s.de/dblp.rdf.gz

[50] DBpedia Dataset. [Online] Available: http://wiki.dbpedia.org/Downloads2015-10

[51] T. Heath and C. Bizer. "Linked Data: Evolving the Web into a Global Data Space", ser. Synthesis Lectures on the Semantic Web: Theory and Technology, Morgan & Claypool, 2011. http://doi.org/10.2200/S00334ED1V01Y-201102WBE001

[52] OpenLink Virtuoso Universal Server. [Online] Available: http://virtuoso.openlinksw.com/

[53] SQLizer. [Online] Available: https://sqlizer.io/#/

[54] Convert JSON to CSV. [Online] Available: http://www.convertcsv.com/json-to-csv.htm

[55] Advanced XML Converter. [Online] Available: http://www.xml-converter.com/features.php

[56] S. Hassas Yeganeh *et al.*, "Linking Semistructured Data on the Web", in *Proceedings of the Fourteenth International Workshop on the Web and Databases (WebDB)*, 2011.

[57] I. Augenstein *et al.*, "LODifier: Generating Linked Data from Unstructured Text", in *Proceedings of the 9th International Conference on The Semantic Web: Research and Applications (ESWC'12)*, Heraklion, 2012, pp. 210–224. http://dx.doi.org/10.1007/978-3-642-30284-8_21

[58] P. Cimiano and J. Völker, "Text2Onto: a Framework for Ontology Learning and Data-Driven Change Discovery", in *Proceedings of the 10th International Conference on Natural Language Processing and Information Systems (NLDB'05)*, Alicante, 2005, pp. 227–238. http://dx.doi.org/10.1007/11428817_21

[59] Integrating Structured and Unstructured Data Using Text Tagging and Annotation. [Online] Available: http://www.bi-bestpractices.com/view-articles/4735

[60] S. Auer *et al.*, "Managing the Life-Cycle of Linked Data with the Lod2 Stack", in *Proceedings of the 11th International Semantic Web Conference (ISWC)*, Boston, 2012, pp. 1–16. http://dx.doi.org/10.1007/978-3-642-35173-0_1

Received: November 2016
Revised: July 2017
Accepted: July 2017

Contact addresses:
Shreyas Suresh Rao
Department of Computer Science and Engineering
Manipal Institute of Technology
Manipal University
India
e-mail: shreyassureshrao@gmail.com

Ashalatha Nayak
Department of Computer Science and Engineering
Manipal Institute of Technology
Manipal University
India
e-mail: asha.nayak@manipal.edu

Shreyas Suresh Rao is a research scholar in the department of Computer Science and Engineering at Manipal Institute of Technology, Manipal, India. He obtained his M.S in Software Systems from BITS, Pilani in 2007. His previous experience includes seven years of working at SLK Software Services Pvt. Ltd., Bangalore where he played various roles such as Business Analyst, Team Leader, Analyst/Designer, and had been involved in the execution of several end-to-end enterprise projects in banking, manufacturing and automobile domains. His technical expertise lies in ASP.NET technologies such as Web services, Web applications, Silverlight, and SharePoint. His research interests include Semantic Web, Linked Open Data, crowdsourcing, knowledge engineering and Web services.

Ashalatha Nayak is a professor and currently heads the Department of Computer Science and Engineering at Manipal Institute of Technology, Manipal, India. She obtained her PhD from the School of Information Technology, IIT Kharagpur in the area of Model Based Testing. Her research interests include Semantic Web, software testing, intelligent agents and cloud security.