Transformation and Annotation of Crowd-Sourced open data into the Global Urban Data Repository

by

Lan Xiao

Supervised by Prof. Mark Fox

April 2018

Abstract

The Global Urban Data Repository (GUDR) is an open crowd-sourced repository of urban data built on the standards of the Semantic Web. To promote data sharing, we relax the data curation requirement in the GUDR. My thesis attempts to design and develop depositor services to automatically translate various input data into linked open data. Since we use crowdsourced data, it is critical to ensure data integrity so that effective decision can be made based on these data. We approach the problem by assessing the quality of data through trustworthiness. We propose a user registration system which verifies authentic users. In addition, we assess the quality of crowdsourced data by formulating an accurate approximation of the trustworthiness of data as well as data providers. Such trust scores represent key information based on which data users may decide whether to use the data and for what purpose.

Acknowledgements

I would like gratefully acknowledge my supervisor Dr. Mark Fox, for all of his assistance with the thesis work and the helpful discussion we have had about this and other topics throughout my fourth year. It was a pleasure being his thesis student. Among others, my research has been aided by communications and conversations with Cheng Lu and Lu Qian. I thank the reviews of my papers for their helpful suggestions and insightful improvements.

Table of Contents

1. Introduction

The increasing urbanization and rising need for sustainable development demand effective urban planning, infrastructure development and upgrades. This is a difficult problem due to fiscal limitations and dynamic nature of cities. A novel approach to address this issue is through urban informatics, which uses urban-related data, with aids from mathematical tools and computer science, to better understand urban systems and make economic decisions. Recognizing the importance of urban informatics, an immediate question to ask is how to discover, organize and make openly available urban datasets, which would remain hidden in the nooks and crannies of the Internet. With the research problem in mind, we propose the Global Urban Data Repository (GUDR), a crowd-sourced open repository of urban data build on the standards of the Semantic Web.

1.1. Semantic Web

To date, the World Wide Web has developed most rapidly as a medium of documents for human consumption. As data is intermingled into the surrounding text, it is hard for programs to process useful information automatically. In addition to the classic Web, the Semantic Web is an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation [20]. To make the web grow, Tim Berners-Lee, the inventor of the World Wide Web, proposed the following rules:

I. Use globally unique URIs to identify online resources;

II. When users look up a URI, provide useful information using the standards such as RDF;

III. Include links to other related URIs to form a web of data;

Some necessary technologies for achieving the functionalities of the Semantic Web includes explicit metadata, ontologies, logic and agents. The main focus of this thesis are on designing and implementing metadata and ontology technologies for the Global Urban Data Repository. In the rest of this subsection, we introduce these two technologies.

1.1.1.Explicit Metadata

HTML is the predominant language in which Web pages are written in. It uses simple structure targeted at human users. A computer often has problem understanding the meaning of the web content and hence making useful decisions. The Semantic Web includes structured information, which is easily processable by machines, to describe the content of Web pages. The term *metadata* refers to such information: data about data.

Resource Description Framework (RDF), is a foundation for processing metadata; it provides interoperability between applications that exchange machine-understandable information on the web. In essence, it is a data model that represents information as node-and-arc-labeled directed graph. The underlying structure of RDF statements is a collection of triples, each consisting of a subject, a predicate and an object. A simple example is :

John Smith	has nick name	Jonny
Subject	Predicate	Object

The subject of a triple uses an RDF URI reference to identify the described resource. The object can be a simple literal value or an RDF URI reference of another resource. The predicate is another URI indicating the relationship between subject and object. RDF is a data model that is independent of any specific serialization syntax, sample serializations includes RDF/XML, RDFa and Turtle.

1.1.2.Ontologies

While RDF augments Web pages to allow more advanced knowledge management systems, problem arises when two database use different identifiers for the same concept. For example, zip code and postal code refer to the same concept in different databases. A solution to this problem is the use of Ontology. In general, an ontology is a set of explicit and formal specifications of a conception [22]. An ontology consists a finite set of terms, and the relationships between these terms. The terms denote important concepts in the domain and the relationships typically include hierarchies of classes. Typically, an ontology for the Web has a taxonomy and a set of inference rules. The taxonomy defines important concepts of the conceptualization, and the inference rules allow a program to deduce new information.

In addition, ontologies may include information such as properties, value restrictions, disjointness statements, and specifications of logical relationships between objects. In the context of the Web, ontology provides a shared understanding of a domain. Such an understanding is necessary to overcome the difference in terminology. Also, ontology helps to improve the quality of web searches.

There is no clear division between what is referred to as vocabularies and ontologies. In practice, ontology is used for more complex and formal collection of terms, whereas vocabulary is used in less restricted ways. In this paper, we use these two terms interchangeably.

1.2. Global Urban Data Repository

The Global Urban Data Repository (GUDR) is an open crowd-sourced repository of urban data built on the standards of the Semantic Web. It uses crowdsourcing approach to promote data sharing, meaning any person or organization may deposit urban related data into the repository. Urban data is defined as anything urban related, such as transportation, governance, social services, education or finance. Our vision is to support the emergence of an eco-system where registered authentic sources, such as city departments, academic research labs, or large corporations will provide a continuous flow of data to the repository. At the same time, we also anticipate less authentic sources, such as individuals and small-scale organizations to deposit data of lesser degree of validity.

A growing number of urban data repositories have emerged over the last five years. Examples include Glasgow's Urban Big Data Centre, Australian Urban Research Infrastructure Network and namara.io. Similar to these repositories, the GUDR aims to discover, organize and make openly available urban datasets. In addition, the GUDR has a set of distinct features [23]. In this paper, we focus on:

A. A single representation is to be used to store all data deposited in the repository, namely RDF triples.

B. All RDF triples will have metadata stored with it to support identifying the data. For example, source, data of creation, data of deposit in the repository, people who deposit the data and trustworthiness.

C. The repository will identify preferred ontologies for resources and properties to be stored, though it is not required to use them.

D. All data will be crowd-sourced in the sense that any person or organization may deposit data into the repository without data curation.

1.3. Problem Statement

To realize the vision of the Global Urban Data Repository, we need to provide tools for the deposition of data into the GUDR. One research challenge is to translate various input formats including but not limited to spreadsheets, JSON, and RDF/OWL into RDF triples. After the data is translated in to a unified format, we want to annotate the data with metadata and vocabulary. In fact, a surprising amount of data does not have explicit metadata or ontology. Instead, ontologies are often encoded implicitly in dataset attributes. In some cases, a dataset uses existing ontologies in their attributes. The generated RDF would be significantly more useful if it maps to the correct existing ontologies. It is also desired that the depositor service provides for the attachment of additional metadata to each dataset for supporting the evaluation of data quality.

The availability of comprehensive urban data makes it possible to extract more accurate and complete knowledge and thus supports more informed decision making. However, reliance on crowd-sourced data for decision making requires data to be of good quality and trusted. While researches have been done to ensure semantic integrity and confidentiality for shared data, the assessment of crowd-sourced data quality remains an open challenge [1]. Our approach is to assess the quality of crowdsourcing data through trustworthiness. Thus, the second research challenge concerns the development of a source registration system that assists data users in assessing the level of trust they should place on a dataset. Since people often recognize datasets deposited by authentic users as more reliable, one problem is how to identify authentic users and verify their expertise. At the same time, expertise is highly dynamic and varies in level [24]. One challenge is how to infer and update user expertise over time to provide accurate and most up to date information. In light of a formal skills ontology developed by Fazel-Marandi and Fox [25], it is possible to model user skills and proficiencies over time by starting with less scrupulous

information about individuals and transforming it into something good and reliable. Hence, the problem simplifies to how to retrieve initial identifying information from users and maintain a continuous flow of information to update users' expertise.

Reputation systems provide a way for updating trust information by utilizing community-based feedback about past experience of users to help making recommendation and judgement on the quality and trustworthiness of datasets and data providers [25]. The challenge of building such a reputation-based trust mechanism is how to effectively cope with various malicious behaviour of users such as providing unreliable feedback about data providers. Another challenge is how to provide direct incentive to raters as there is no rational reason for providing feedbacks.

With the proposed research problem in mind, we develop the GUDR depositor service, a three component solution to facilitate data curation and data quality assessment in the GUDR. The three components include a data depositor, a source registration system with optional user verification, and a reputation-based trust supporting framework. Combing the second and last component, we aim to build a rating aggregation community that provides the following value-added information to users:

I. User profiles declaring skills, skill proficiencies, degrees, and experience;

II. Confidence level of datasets and data providers in the form of trust scores;

III. Feedbacks from previous users in the form of text reviews.

1.4. Contributions

The high level objectives of this thesis revolves around solving the problem of data curation and data quality assessment in the Global Urban Data repository. We summarize the specific algorithms and systems developed during this thesis as follows.

- Algorithms to translate datasets into RDF triples, define explicit vocabulary and provide for the attachment of metadata;
- A general framework for verifying authentic users;
- A method to compute quantitative measure of the trustworthiness of datasets and data providers.

1.5. Organization

The rest of the paper is organized as follows. Section 2 provides a literature review that summarizes and synthesizes relevant research in constructing an understanding of the current state of crowdsourced data curation and quality assessment. Section 3 introduces and justifies the methods and design decisions that have been chosen in this project, namely the depositor algorithms, the registration system and the reputation-based trust model. Section 4 describes a set of experiments that are carefully designed to evaluate the depositor service by showing its effectiveness, robustness and efficiency. Section 5 concludes the paper with a few directions for future works.

2. Literature Review

2.1. Crowdsourced Data

The advance of Web facilitate the collection of information and distribution of problem solving in new ways. This crowd-sourced data can be leveraged to benefit the society if it is appropriately processed and analyzed. However, this requires data to be of good quality and trusted. Data quality in professional database is guaranteed by certified authorities. The crowdsourced data, on the other hand, lacks such guarantee. Due to the nature of crowdsourced data, a quality assessment step is necessary for the data to be ready for use. However, the assessment of crowd-sourced data quality remains an open challenge [1].

The quality of data can be compromised in various of ways, including data tampering by malicious parties, inaccurate or incorrect recordings, and data deceptions. A basic approach to assess data quality would be comparing it with a professional dataset, which is taken as the ground truth answer. This approach makes implicit assumption that such datasets exist and are accessible. However, this assumption is generally not true. The ground-truth datasets are expansive, if not impossible to access. A different approach as suggest in [1] consists in assessing the quality of crowdsourcing data through trustworthiness, which is defined as ‘bet about the future contingent action of others’ [2]. This definition emphasizes on establishing trust and developing systems that can assist data users in assessing the level of trust they should place on a dataset. To validate the trustworthiness of crowdsourced data, various techniques could be

employed. Voting techniques are proven to be successful in determining the trustworthiness of messages from various social media sties. For example, YouTube users can provide binary feedback for user comments. As a result, comments with too many negative feedbacks are hidden [27]. Other techniques, such as reputation and trust modelling can also be useful in determining the trustworthiness of crowdsourced data.

This literature review serves as a critical summary of published literature relevant to the curation and trust assessment of crowd-sourced data. The rest of this subsection is organized as follows. Section 2.2 introduces research challenges revolving around data quality that are encountered during data sharing. Section 2.3 presents the current state of the art information integration technologies builds on the standards of Semantic Web. Section 2.4 summaries prevailing methods adopted by online communities to verify authentic users. Section 2.5 introduces different approaches to evaluate the trustworthiness of both data and data providers. Section 2.6 concludes the review and outlines the future research directions.

2.2. Research Challenges

Crowdsourcing is an online production model which has increasingly gaining attention recently. The availability of comprehensive data generated by crowdsourcing make it possible to extract more accurate and complete knowledge and support decision making process. The anonymity of crowdsourcing, however, introduces possible inaccurate or incorrect data. Some common problems reported from surveys in [11], [12] are summarized:

- Computer mediated community involves a large number of participants with different social and professional background. Without knowing the identities of data providers, data users cannot recognize source credibility.
- Taking the advantage that reputation can be computed implicitly from ratings, most existing rating systems are not capable of differentiating good feedbacks from malicious ones. This introduces vulnerability to the rating system under dishonest feedbacks.
- Reputation should be context sensitive. For example, experts on electricity should have high reputation for electric data, but not necessarily for geographic data. However, most systems do not provide support to incorporate context information.

- Most systems provide no incentives for users to give reviews. This lead to insufficient feedback.

In the following sections, prevailing models for corresponding online identities to offline identities are presented; literatures revolving around building trust and reputation systems for online communities are reviewed.

2.3. Information Integration

Linked data continues to grow at a rapid rate, but often a meaningful semantic description is missing from most data publishes today. This lead to a limitation in the expressiveness of linked data. The problem of data integration is to combine data of different sources and provide users a unified view of the data. In this subsection, we present tools, such as D2R [28], which enable a user to translate a dataset into RDF effortlessly. The restriction of such tools lies on the lack of ability to provide support to easily map the data into an existing ontology. Therefore, more advanced tools, such as Karma [29], which allows a user to map structured source to ontologies in order to build semantic descriptions, are included to complete the understanding of the state of the art in the field. In the end, a clear research gap is identified to justify the purpose of our research.

- *D2R Server:* D2R Server is a tool for publishing the content of relational databases on the Semantic Web. Specifically, data on the Semantic Web is modelled and represented in RDF. D2R Server uses a customizable D2RQ mapping language to map data content into RDF, and allows the RDF data to be searched and browsed. D2RQ is a declarative language for mapping relational database schemas to RDF vocabularies and OWL ontologies. D2RQ allows users to define a mapping file with rules to establish a RDF vocabularies that maps RDF classes and properties to database tables and columns. D2R server only provides RDF interface for relational databases.
    - *Karma:* In addition to D2R Server, Karma is an extended information integration tool that defines the contents of a dataset in terms of a given ontology. Users can define their own ontology or bring in an existing ontology that may already have been used to describe other related datasets. The advantage of Karma is that it is possible to generate RDF triples with

respect to a specific domain knowledge. Karma converts source data into RDF with explicit underlying ontology, but does not provide the attachment of additional metadata. Users also need to provide the correct ontology which they wish to use.

The research objective is to relax the data curation requirement for the GUDR. We attempt to have a set of tools that automatically translate various input data types into RDF triples and deposit them into the repository. A key functionality which is missing from the above tools is the ability to provide the attachment of additional metadata to support identifying the data more accurately and precisely. In addition, vocabularies/ontologies should be automatically generated to specify the domain knowledge. In cases where existing ontologies are used in the source dataset, the correct mapping should be automatically established to make useful links to other related datasets.

2.4. Verify Authentic Users

Many online communities provide verification for authentic public figures and organizations. Different badges are often used to differentiate different types of public figures. In this section, several models will be examined and compared. Potential improvements are identified.

- *Facebook:* Two types of verification status are used, including blue checkmarks and grey checkmarks. The former are given to public figures, media companies and large brand [13]. The grey checkmarks are given to smaller or local business and organizations. For example, specific locations of a large company are eligible for grey checkmarks[14]. Blue checkmarks are given to both Pages and personal profiles, while grey checkmarks are only given to Pages. After users submit request for verification, Facebook will call the Page's publicly listed phone number to provide a verification code. After the users enter the code, Facebook will proceed to verify the Page manually based on requestors' Facebook profile. One important factor that determines if the users/Pages can receive verified status is the number of followers the users have.
- *Twitter:* Only one type of verification is used to indicate that a twitter account is authentic [15]. An account can be verified if it is of public interest, including celebrities in music, acting, fashion, government, politics, religion, journalism, media, sports, business, and other key

interest areas. To verify their accounts, users need to provide at least 2 websites which prove one's offline identity. Also, the Twitter profile is an important factor in the verification process.

- *Google My Business:* Google My Business is an online dashboard designed to streamline the management of business information across multiple Google services. Users can add business information to Google Maps, Search and other Google properties by using Google My Business. To verify the ownership of the business, Google currently provides four different methods, including mail verification, phone call, email verification, and video verification. Among the above methods, phone call and email options are only given to some large businesses. Video verification is only offered in select regions where mail verification is uneasy, which is currently only piloting in South America and parts of West Africa. For mail verification, a postcard with verification pin will be mailed to the specified business address. Google also provides mechanisms to request access to your business if it has been claimed by another account. This is an important feature but is missing from most online communities[16].
- *Alibaba:* Alibaba is an online eCommerce community. To become a verified seller, Alibaba requires the sellers to upload a piece of legal photo identification. After the ID has been verified, the sellers are further required to upload a picture of themselves holding the photo identification which is used in the previous step.

Identity verification is not a trivial problem. Current solutions mainly involves manual verification supervised by specialists. The prevailing mechanism is verification against email domain name. For added confidence, websites and phone numbers are used to relate one's offline identity to online identity. This provides enhanced trusts and should be indicated by different verification status. For example, if a user provides an email with correct domain address, a public listed phone number and a website that proves his or her offline identity, the user receives a higher level of trust than a user who only uses the correct email. Google My Business allows multiple users to be listed as the owners of businesses while all of the other models summarized above do not provide the ability to register multiple accounts under the same organization. This is an important feature for the GUDR in order to support the crowdsourcing approach. Individuals of the organization are encouraged to participate rather than a single representative

who speaks for the organization. None of the above models builds an internal representation of the organization hierarchy.

2.5. Building Reputation and Evaluating Trustworthiness

Identity is formed in the process of categorizing, classifying or name oneself in particular ways in relation to other social categories or classification [3]. In social identity theory, a view of the group as a basis for identity is primarily adopted (who one is). In identity theory, a view of the role as a basis for identity is used (what one does)[4]. For our purpose of identifying one's reputation, we consider both the role and the group bases of identity to reinforce who one is. It is pointed out in [3] [5] that self verification promote participation in social movements.

Credentials are issued by the same kinds of organizations that issue paper credentials today. For security purpose, credentials are typically created offline and then either securely distributed to their new owners or made available for pickup in a semipublic database. Each issuer can use locally created identities to refer to the parties mentioned in its credentials, rather than a globally unique identity that would allow easy tracking of the parties' activities[6]. A credential verification system would be objective and similar in principle. It aims to correspond online identities to offline identities associate with real people and selectively disclose credentials. Such a system could potentially encourage greater accountability for users who claim expertise in certain fields. It helps to build the GUDR's creditability, promotes extended confidence from data users, and fosters credit between fellow data contributors.

The effective identity communication is underscored in Goffman's self-presentation theory that argues people desire to explain themselves to others regarding their identities before concentrating on work or other goal that bring them together. By reaching a consensus regarding identities, people fell understood and obtain a sense of continuity and coherence [7]. In a virtual environment setting, people help strangers not only because of altruism, but for reputation [8], future reciprocation [9], and self-esteem [10]. Therefore, online identity provides significant motivation for knowledge contribution.

Several reputation systems and mechanisms have been proposed for online environment. The most popular trust management technique in such system is to compute the trust score for each

data provider based on ratings. PeerTrust [17] is such a reputation-based trust supporting framework in peer-to-peer e-commerce communities. It includes five trust parameters in computing trust score of data providers, namely, feedback a peer receives from others, the total number of transactions a peer performs, the credibility of the feedback sources, transaction context factor and the community context factor. Instead of summing directly over all the feedback a data provider receives, which has been proved by Dellarocas [18] that it does not function well and the resulting prediction outcome will be unfair, the framework calculates a weighted average of all the feedback received by a data provider. The weights are assigned based on the credibility of the raters, where rating from those raters with high credibility are weighted more than those with low credibility. To calculate the credibility of the raters, the framework proposed two methods. The basic approach is to use a function of the trust score of a user as its credibility factor. This is a computationally inexpensive method which depends on two assumptions, namely untrustworthy users are more likely to submit false retires and trustworthy users are more likely to be honest on the ratings they provide. Alternatively, a personalized similarity measure can be used. Based on the review history of a user *u*, this metric calculates the similarity between *u* and other raters. High similarities corresponds to large weights. In addition, a transaction context factor is used to incorporate the information on the scope of the transactions, and the context of the transactions. Lastly, community context factor is used to distinguish expertise in different sectors. Experiments have shown effectiveness of PeerTrust in P2P communities.

However, PeerTrust only computes the reputation of participating parties in the system, where there are more influential factors affecting the trustworthiness of data. In [19], an approach to evaluate data trustworthiness based on data provenance is proposed. It takes into the account of data similarity, data conflict, and path similarity to assign trust scores to both data and data providers. This approach uses an iterative method to compute the trust scores. To start the computation, each data provider is first assignment an initial trust score. At each iteration, the trustworthiness of the bata is computed based on the combined effects of the aforementioned three aspects, and the trustworthiness of the data provider is recomputed using the trust scores of the data it provides. A clustering algorithm is used to relax the computational burden, where

similar data is clustered and treated as a single entity. In particular, data similarity is calculated based on the SSD between two numerical values or the hamming distance between two strings or categorical values. The path of a data is defined by the source provider and a set of intermediate agents that processed the data. Based on the reasonable assumption that the probability of multiple source providers and intermediate agents reporting the same wrong information is lower, the less correlation among data generation path of the same data, the more trustworthiness the data is. Data conflicts refers to inconsistent descriptions about the same event. Since it largely depends on the knowledge domain of the specific application, the model provides the flexibility for users to define their own data conflict functions. After computing the three factors influencing the trustworthiness of data, the overall trust scores is computed iteratively. The iteration stops when the changes to trust scores become negligible.

The above two models shares some similarities but differ in their fundamental ideas. Both models attempt to calculate trust scores to estimate reputation and trustworthiness. However, PeerTrust concerns the reputation of participating parties in the community while the data provenance trust model studies mechanisms to evaluate trustworthiness of data. In addition, PeerTrust evaluates data quality based on the trustworthiness of the data provider, while the data provenance trust model investigates other important factors involve in the process of data collection and generation, data similarity and data conflicts. The second approach is computationally more demanding. For a crowdsourcing data repository, the size of the database can grow indefinitely. Hence making the second approach hard to maintain. On the other hand, the first framework better exploits the virtue of crowdsourcing.

2.6. Conclusion

In this section, different approaches which attempt to evaluate the trustworthiness of crowdsourced data are reviewed. Although this review cannot claim to be exhaustive, it does provide reasonable insights and shows the incidence of research on this subject. Four verification methods used by major online communities are presented. Review of literatures also shows that identifying data providers promotes knowledge contribution in online community. Two models aiming to evaluate trustworthiness are reviewed. While the first model attempts to evaluate the

reputation of users, the second model incorporate more important factors concerning data quality. In the future, we plan to integrate the two models and implement a viable solution specific to the GUDR to better evaluate the trustworthiness of crowdsourced data. A framework to build internal representation of the organization hierarchy is also needed.

3. Data Curation and Quality Assessment

Our approach to solve the data curation and data quality assessment problem in the GUDR is an integrated solution consisting three components, namely a data depositor, a source registration system and a reputation-based trust supporting framework. In this section, we present the design and implementation for each of the three components. Specifically, the data depositor provides data curation services to Excel spreadsheets, CSV files and JSON files (Section 3.1). The source registration system allows users to build their profiles in the GUDR, which prepares the initial information needed to build a knowledge base using *Ontology of Skill and Competency* (Section 3.2). The Ontology is further exploited to reason about user's skills and competencies in a dynamic environment. We then describe a reputation-based trust supporting framework. It uses an iterative method to compute a quantitative measure of the trustworthiness of users based on crowdsourced feedbacks (Section 3.3). This provides a continuous flow of new information to either confirm or update what is already believed about a user's skills.

3.1 Modeling Structured Inputs

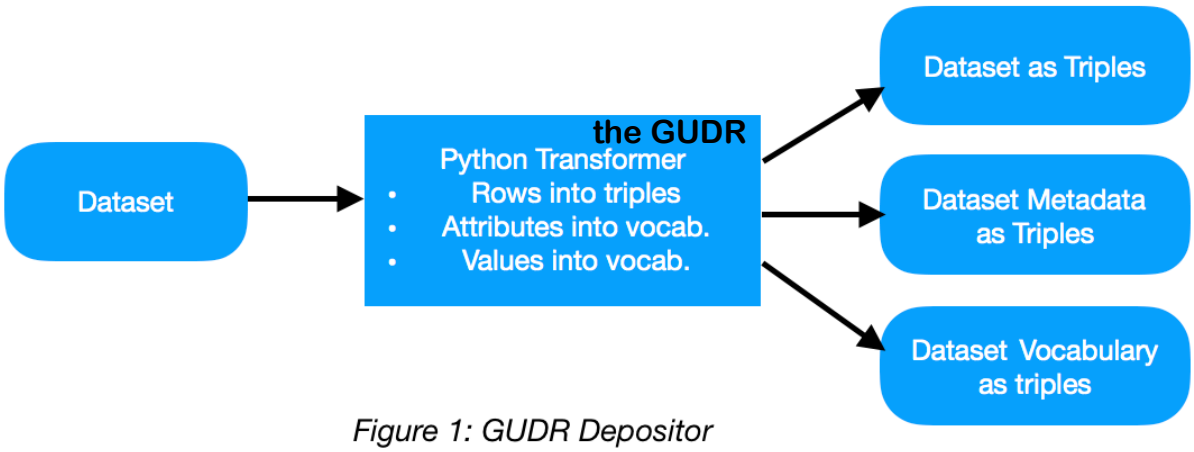


Figure 1: GUDR Depositor

Figure 1 illustrates our approach for translating data in structured sources to RDF expressed in terms of a vocabulary automatically generated from dataset attributes. The input to the depositor is a collection of datasets in the supported formats. The outputs of the process include a refined dataset in RDF triple format, a domain specific ontology based on user specification, and a structured metadata to describe the dataset and data provide.

The data depositor process the input datasets in four steps. The first step, *extract dataset vocabulary,* extracts domain specific information from each column of dataset attributes and map the information to a node in the vocabulary. The vocabulary file is published into the Linked Data cloud; it is used to classify the terms that can be used in the given dataset, characterize possible relationships, and define possible constraints on using those terms. The data depositor provides a command line interface to let users assign semantic types such as language tags to the columns of a dataset attributes. This is optional, and without user input, the default semantic types include a rdf:type predicate represented by the IRI http://www.w3.org/1999/02/22-rdf-syntax-ns#type and a rdfs:label predicate represented by the IRI http://www.w3.org/2000/01/rdf-schema#label .

The second step, *translate data entries,* involves translating each data entry into a series of RDF triples using the user defined vocabulary. For an Excel spreadsheet or a CSV file, we require input file to list all attributes in the first row, each row of a dataset is regarded as a single data entry and is identified by a subject term. Each non-empty column is translate into a RDF statement with that subject. The attributes are the predicate terms and the values are the object terms. For a JSON file, each object is regarded as a single data entry and the key/value pairs defines the predicates and objects with that subject. The depositor is able to assigns types automatically based on the data values in each column and additional user inputs, if provided. The generated RDF file can be previewed before being deposited into the repository. If the semantic type assigned by the depositor is incorrect, the user can correct the translation by restarting the translation process and providing additional information.

The third step is to *create metadata.* This relies on the additional information about the data provider and about the dataset. Such information is provided by the registration system. More

details on the registration will be elaborated in section 3.2. Specifically, the depositor tries to identify unique ID of the file, date of curation, date of data deposited, URI to the file, data provenance and data provider. The deployed vocabulary for representing such data is Dublin Core [21], particularly the `dc:creator, dc:publisher` and `dc:date` predicates. When using the `dc:creator, dc:publisher` properties, the GUDR URIs identifying the creator and publisher are used to refer to the correct users instead of simple literal strings. This allows others to unambiguously refer to them and, for instance, connect these URIs with additional information about them which is available on the Web to assess the quality and trustworthiness of published data. Each metadata file can be uniquely identified and a one-to-one mapping exist to correspond the metadata to the belonging dataset.

The software is written in Python, because the Python language has extensive supported libraries, which includes libraries that perform operations like writing and reading files in different formats. Python can process XML and other markup languages easily as it can run on all modern operating systems through same byte code.

3.2 Designing User Registration System

One of the main focus of this thesis is the design of a systematic procedure to register users into the GUDR and formulate an accurate approximation of the trustworthiness of data as well as data providers. In this subsection, we present a complete user registration system with optional verification mechanism targeted at authentic users. Figure 2 illustrate the complete workflow of registration process. Our framework starts by registering users with valid emails, which do not have to be institutional emails. This relaxation accommodates less professional sources, such as individuals and less scrupulous organizations to share information of less degree of validity. A confirmation email is sent out to complete the registration. Upon successful registration, a user profile will be created in the GUDR database to track a user's skills, proficiency levels, degrees, and experience. Formally, a *skill* suggests the possibility of performing an activity; *proficiency level* refers to the ranking of the ability of an individual to perform the activities enabled by a particular skill. This information can be used to accurately model user expertise that is domain specific. For example, John declares he has the database design skill at level 2 when he

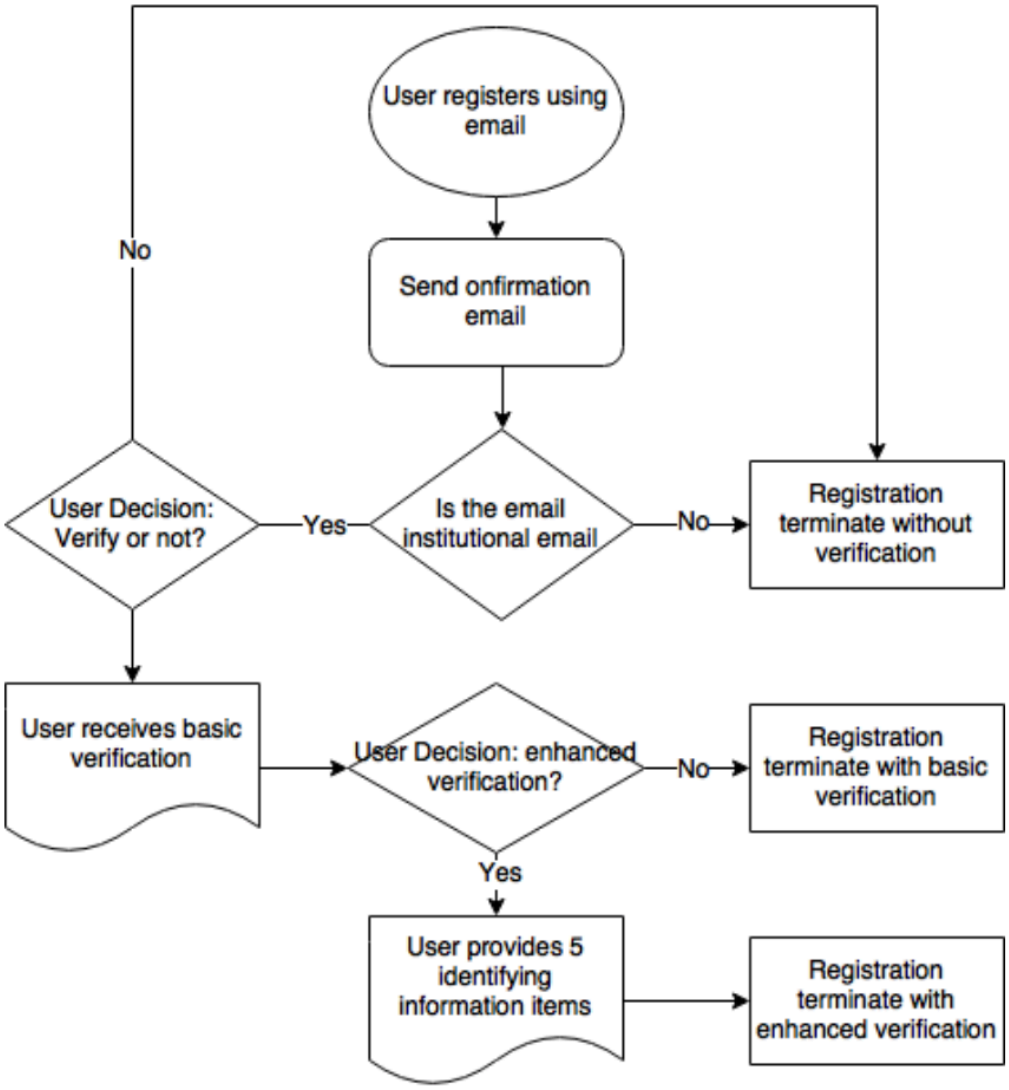


Figure 2: Workflow of registration process

registered in the GUDR. Since this is a self-declaration, we can only recognize the information with limited confidence. However, if we have the additional information that he has previous experience as database engineer which requires database design skill at least at level 2 and he excel in his job, we can infer that most likely he has database design skill at level 2. As more information becomes available on activities performed by John, what is known about him can be confirmed, refuted, or revised. In addition to the activities that one can perform, the registration system also considers attributes related to a skill that are measurable. For example, years of

experience, the time it takes to complete the activities, and syntax error found in the deposited datasets are some of the attributes that can be used to measure proficiency in database design. Hence, providing details about themselves facilitate the modelling of user expertise in the GUDR but is not mandatory. In addition to expertise modelling, user profiles are also used later to formulate metadata associated with data deposited into the GUDR by the corresponding data providers. Completing the user profile also contribute to the verification step, which will be elaborated next.

User verification is a separate process from user registration and is an optional step for providing enhanced confidence in data providers. But unverified users can also deposit data into the GUDR. Our framework does not offer automatic verification, a user need to submit verification forms to apply for a verified status. In our framework, two levels of verification are employed and different badges are used to make the distinctions. For basic verification, it is required that users register with institutional emails. The institutional email address indicates a connection between the user and the institution, but more information is required to ascertain the type of connection and hence determine the level of trust. For example, in a university setting, both a student and a professor have a university email address, but the information provided by a professor should receive more attention and the information provided by s student should be taken with a grim of salt. The enhanced verification requires the users to provide more information about themselves. We identify five important items for enhanced verification:

1. The name of the data provider
2. The current affiliation
3. Official titles
4. At least two publicly accessible websites that can prove the data provider's offline identity
5. A public listed phone number

To illustrate, we requires at least two publicly accessible websites that can prove one's offline identity. We also use public listed phone number to relate one's offline identity to their online identity.

Users are also encouraged to provide details about the organization or company they worked at to help construct the GUDR internal representation of the organization or company. For example, again using the university setting, a professor who is affiliate with the MIE department at the University of Toronto also runs his own research lab. He can help the GUDR build a hierarchical representation of the University of Toronto by declaring his research lab as a subclass of the MIE department, and the MIE department as a subclass of the University of Toronto.

The process of verification is a manual process, involving looking up the domain address of the institution or organization, calling the registered phone number, verifying the registered phone number against the public listed number for the institution or organization, and browsing through the reported website to confirm user's offline identity. The above information is included in metadata, hence the data deposited by authentic users inherent the verified status. A logical theory for trust in the form of ontology has been developed in [30]. The ontology defines formal and explicit specification for the semantics of trust. From this formal semantics, two types of trust: *trust in belief* and *trust in performance* are identified. In the context of the GUDR, we deal with *trust in belief*. It has been proved that trust in belief is transitive. Because of this property, trust can propagate from data providers to the datasets deposited by the provider. Hence, it is formally proved that datasets deposited by trusted users are more reliable and have better quality.

The framework endeavour to fully exploit the power of crowdsourcing, hence users can report on bogus information about others in the form of text reviews. If a verified user receives a significant number of reports, the framework will review the user's profile and potentially remove the verified status. More importantly, the text reviews are available to the potential users of a dataset. It is up to the users to use their discretion in deciding what is the minimum trust a dataset need to have in order to fulfill their specific need. The verification sets a benchmark for determining the reputation of users. It serves as an indication of the authenticity of a data provider, which helps data users to infer the trustworthiness of data providers.

3.3 Reputation-based Trust Supporting Framework

Ratings for datasets and data providers are important for the GUDR, as they allow users to harvest the wisdom of the community in making decision. Therefore, a reputation-based trust

supporting framework is developed. It uses a crowdsourcing approach for assessing how trusted the data is, based in turn on the feedbacks the data provider receives in depositing other data into the GUDR in the past. Such feedbacks reflect the degree of trust that other users in the GUDR have on a given data provider base on his past experience. However, the difficulty with feedbacks is that little is known about the users providing them. Interpreting the ratings well requires that the reputations of raters be factored into the scores computed for rated objects, even though these reputations are not explicitly available. Instead, the reputation of raters can be computed implicitly from ratings. In the rest of this subsection, we present design details of the framework and strategies for developing system-level mechanisms to implement the proposed model.

To calculate the reputation, a rating aggregation system is designed and implemented based on PeerTrust framework[17]. Details of the original framework is presented in section 2.4. We have adapted the framework for our purpose. We identify four important factors for evaluating the reputation of a user (Section 3.3.1), formalize these parameters to present a general trust metric that integrates these parameters in a logical procedure (Section 3.3.2), and describe the formula we use to calculate the values for each of the parameters in the setting of the GUDR (Section 3.3.3). We discuss the dynamic nature of user expertise in Section 3.3.4 and address such potential dynamic behaviours. We conclude the section with discussion on feedbacks in terms of text review and how to incorporate them in the system (Section 3.3.5).

3.3.1 Trust Parameters

We introduce four important parameters for evaluating a user's reputation:

1. The ratings a user receives from other users
2. The feedback scope, such as total number of downloads for the data provided by a user
3. The credibility for the source of feedback;
4. The community context factor for addressing some characteristics and vulnerabilities in the GUDR.

Comparing to the original PeerTrust framework, a transaction context factor for discriminating important transactions from less or noncritical ones in the original framework is removed, since this does not apply well to urban datasets. We illustrate the importance of the above four parameters.

- *Feedback in terms of Numerical Ratings:* After a user download a dataset, GUDR promotes the user to submit review to reflect how well this dataset has fulfilled the user's need. Many different feedback format exist, such as positive feedback, negative feedback, a numeric rating or a mixed format. A data provider's rating is a function of all the ratings he receives from all datasets he has deposited into the repository. In some traditional reputation systems, this is the sole parameter being used. To compute a user's trust score, a direct summation of all the ratings a user receives is used. The problem with these systems is that a user who deposits little high quality data will be considered as not trustworthy when comparing to a user who deposits a large amount of data of less degree of trust. In general, binary reputation mechanisms will not function well and the resulting output will be unfair if judgement is inferred from knowledge of the sum of positive and negative ratings alone [31].
- *Feedback Scope:* An important measure which reflect the feedback scope is the total number of reviews a user receives from all datasets he deposited. As mentioned above, using a simple summation of ratings to calculate trust score does not generalize well when the number of datasets deposited differs substantially from user to user in the repository. To eliminate the effect of different scopes, the metric to calculate trust score should use a normalized numerical rating for each data provider, i.e., the average rating of a user as defined by the ratio of the basic summation of all ratings a data provider receives over the total number of reviews the data provider has. We expect people to favour data deposited by verified users, hence the scope factor implicitly provides incentives for verified users. This encourages people to fill in their profiles and verify their authenticities. However, the reliability and quality of these reviews are not well defined. When considering reputation information, we need to account for the source of informational as well as the context.

- *Credibility of Raters:* A feedback is a statement from the data user regarding how satisfied they feel about the quality of the dataset provided by the data provider. A data user may make false statements about one or more data providers' datasets intentionally or accidentally. In either case, this tampers the data provider's reputation unfairly and a trustworthy data provider may end up getting unreasonable trust score even if he has always been depositing good quality data. Therefore, we include the credibility of the raters as an important factor when calculating the trust scores for the GUDR users.
- *Community Context Factor:* This incorporate some of the community-related issues in the calculation of trust scores. For example, this provides a default score for new users who do not receive enough feedback for it to be meaningful. The default value, in term, based on whether the user is verified in the GUDR or not. As another example, we use the community context factor to reward users who submit reviews. This can mitigate the problem of not enough feedbacks are collected to calculate trust scores.

3.3.2 Defining Trust Metric

After defining the four important parameters, we need to combine these parameters into a trust metric in a logical manner. A mathematical formula to compute the values for each parameters given a data provider will be described.

Let $D(u)$ denote the total number of datasets that a user u has deposited into the GUDR, $F(u,i)$ denotes the total number of feedbacks that user u receives for his ith dataset, $r(u,i,j)$ denotes the jth raters who writes reviews for user u's ith dataset, and $S(u,i,j)$ denotes the normalized rating user u receives from $r(u,i,j)$ regarding his ith dataset, $Cr(v)$ denote the credibility of the user v as a rater, and $CF(u)$ demote the community context factor for user u. Then, the mathematical formula to represent the trust score of user u denoted by $T(u)$ is defined in (1).

$$T(u) = \alpha \times \sum_{i=1}^{D(u)} \left[\sum_{j=1}^{F(u,i)} \left(S(u,i,j) \times Cr(r(u,i,j)) \right) \right] + \beta \times CF(u) \quad (1)$$

Where α and β indicates the weight factor for the weighted average rating and the community context factor.

There are two components to this trust metric. The first part is a weighted average of all ratings a data provider receives. The first summation outside the square bracket iterates through all datasets a data provider has deposited. The second summation outside the regular bracket iterates through all the ratings a data provider receives for the *i*th dataset. Combing the two summation, we are able to iterate through all ratings *u* receives. The rating $S(u, i, j)$ has been normalized before plunging into the formula, so it combines the first and second parameter. The weight takes into account the credibility of the feedback source to ameliorate dishonest feedback. This approximates the likelihood for data provider *u* to continue deposit datasets of good quality into the repository. The second part of the metric modifies the first part by increasing or decreasing the trust score based on the community context factor. For example, number of feedbacks data provider *u* submit as a rater for other users in the repository increases *u*'s trust score as a data provider. The weight factors α and β control the balance between evaluating one's trustworthiness based on crowdsourced feedback system and on their offline identity. For example, if not enough feedbacks are presented, β should be set to a large number and use one's offline identity to infer the trustworthiness. Notice that these are the hyper parameters to the system. The trust metric may behave differently depending on the value of these hyper parameters. The question of how the parameters should be set is not trivial and requires careful tuning.

3.3.3 Computing Trust Scores

As shown in (2), we first consider the weighted average of ratings a data provider *u* receives from all his datasets. This is equivalent to setting $\alpha = 1, \beta = 0$ in (1).

$$T(u) = \sum_{i=1}^{D(u)} \left[\sum_{j=1}^{F(u,i)} \left(S(u, i, j) \times Cr(r(u, i, j)) \right) \right] \quad (2)$$

Our system uses a dataset-based feedback system, meaning the feedback is associated with each dataset. The system solicits feedback after each download and the data user optionally gives feedback to the dataset based on their satisfaction about the dataset. In GUDR, we use a numeric rating feedback system to estimate trustworthiness. $S(u, i, j)$ is a normalized rating ranging

between 0 to 1 that can be computed based on the feedback. Both the ratings and the number of feedbacks are quantitative measures and can be collected by the system automatically.

On the other hand, the credibility of feedback source is a qualitative measure and needs to be inferred based on past behaviours of a user as rater. One way is to create a separate measure for the quality of feedback, namely feedback about feedback. This is a common approach employed by many P2P eCommerce communities where users tend to behave maliciously by providing fake or misleading feedback about other users more often as there exist a direct competition between them. However, this approach makes the system more complex. In our framework, we use a simple approach to infer the credibility value of a user implicitly by using a function of the trust value of a user as its credibility factor. So feedback from trustworthy users are considered more credible and weighted more than those from untrustworthy user. This approach is sufficient for the GUDR because we expect less malicious review from data users comparing to P2P eCommerce communities. This approach based on two underlying assumptions. First, untrustworthy users are more likely to submit low quality reviews. Second, trustworthy users are more likely to provide true and accurate description about the datasets they use. These two assumptions are reasonable in the GUDR. Therefore, we write a recursive formula that uses the trust score of a rater as his credibility measure and rewrite (2) in the following form:

$$T(u) = \sum_{i=1}^{D(u)} \left[\sum_{j=1}^{F(u,i)} \left(S(u,i,j) \times \frac{T(r(u,i,j)}{\sum_{k=1}^{F(u,i)} T(r(u,i,k)} \right) \right] \quad (3)$$

Most websites that collect information quality ratings do not provide direct incentives to raters. Therefore, the motivation of the raters to provide high quality feedback is a fundamental problem as there is not rational reason for providing ratings, and a potential for free-riding by letting the others do the rating [32]. Some existing online communities use monetary incentive as a reward for users who submit reviews. This is not applicable in our application. Other approaches have been suggested for the incentive problem of reputation system in [26] such as market-based approaches and policy-based approaches in which users will not receive rating information without paying or providing ratings. Implementing these approaches might discourage people from depositing data into the GUDR and hence counter the original purpose of this project. To

encourage users to submit reviews, our framework provides incentives to users who write reviews through community context factor. We define an adapted metric in (3) with a reward as a ratio of total number of feedback a data provider u gives others, denoted as $E(u)$, over the total number of datasets u deposited during a recent time window, denoted by $D(u)$. The weight factor β can be tuned to control the amount of reputation that can be affected by providing ratings to others.

$$T(u) = \alpha \times \sum_{i=1}^{D(u)} \left[\sum_{j=1}^{F(u,i)} \left(S(u,i,j) \times \frac{T(r(u,i,j)}{\sum_{k=1}^{F(u,i)} T(r(u,i,k)} \right) \right] + \beta \times \frac{E(u)}{D(u)} \quad (4)$$

The community context factor also recognizes users with verification status and incorporates this community context factor into the calculation of trust score.

3.3.4 Inferring and Updating User Expertise

Up to this point, the proposed trust metric is independent of its implementation. However, the effectiveness of the supporting trust in the GUDR depends on the specific implementation. Here we present the algorithm and design considerations in implementing the trust evaluation component for the GUDR.

The trust metric we present in the previous sections looks at the recent behaviour of a user to determine the trust score. Recent behaviour is defined as dataset deposited and rating received within last six months. Only a recent window is used instead of the entire history because user expertise is highly dynamic. For example, users may improve their expertise in certain field by obtaining a new degree from a certified institution, or by obtaining additional experience that requires and practice certain skill. Using the trust metric as defined in (4), we develop algorithms to compute and update user trust scores based on the reputation data that are collected in the GUDR. Our algorithms use trust data collected at runtime time to compute the trust value. Since (4) defines a recursive function that uses the trust score of a rater to measure the feedback credibility of that rater, data provider u needs to recursively compute all his raters' trust scores in order to compute data provider u's trust score. Hence, iterative algorithms are used to implement the dynamic computation. Given a community with N users, each can be a data provider as well

as a rater, the algorithms first construct a trust vector of size N and initializes the values to defaults. Again, the default values depend on whether the user is verified or not. As the data provider *u* receives rating from a rater with in the recent time windows, the algorithm repeatedly computes and updates the trust vector until the change between two consecutive computations give similar results. In other words, when the trust computation converges. After each computation, all the users in the community have up-to-date trust scores. The pseudo code is given below:

```
UpdateTrust(u)
Input:u; Output: T(u);
for i from 1 to N:
do
     Retrieve all feedbacks in recent time window for user i;
     T_0(i) = T_default
end For
while δ < ϵ:
     for i from 1 to N:
     do
          T_t+1(i) = Update trust score based on basic metric;
     end for;
     δ = ||T^t+1 − T^t||
end while;
```

3.3.5 Text Reviews

Besides numerical trust scores, a data user can submit a comment in the form of text review about a specific dataset after each download. The review will be available to future users. This provides more semantically meaningful descriptions to the dataset, such as in why is the data satisfying, what are some of the limitations of a dataset, or context information that is missing from the original metadata.

4. Experimental Results

We performed a set of experiments to evaluate the feasibility, effectiveness and benefits of the three-component solution aiming to solve the problem of data curation and data quality evaluation in the Global Urban Data Repository. The first experiment evaluates the functionality of the data depositor using a sample Excel spreadsheet with a single worksheet. The second experiment demonstrates the effectiveness of the user registration system for providing initial information to build a knowledge base. Following that, the analysis of our trust evaluation component is presented to show the effectiveness against false feedbacks and low quality data.

4.1. Data Depositor

There are two main objectives of this evaluation. First, we want to assess the ability of our approach to produce the three required outputs. Namely, a structured dataset in RDF, a vocabulary describing the domain-specific knowledge, and metadata to annotate the dataset with additional information. The second objective is to measure the effort required in our approach to create the required outputs. We quantified the required effort in using our solution by counting the number of user interactions during the translation process that the user had to perform. Since it is intuitive to expand the evaluation to larger files, and the extra entries in the input dataset does not add value to the evaluation in terms of our objectives, a simple Excel spreadsheet with 3 attributes and 4 rows is used to demonstrate.

ID	Title	Year Of Release USA
1	The Matrix	1999
2	Star Wars	1977
3	Aliens	1986
4	Dope	2015

Table 1. Sample input dataset in excel

Table 1 shows the sample input used in the evaluation, this is the input of the data depositor. We require the input dataset to have its first row containing all the attributes, and starting from the

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix dc: <http://purl.org/dc/elements/1.1/> .
@prefix data: <http://exampleGUDR.org/data/myturtle.xlsx/Sheet_1#> .
@prefix vc: <http://exampleGUDR.org/voc/myturtle.xlsx/Sheet_1#> .

<http://example.org/data/your_csv_file.csv#row=1>
        :ID "1.0";
        :Title "The Matrix"@en-ca;
        :Year_Of_Release_USA "1999.0".
<http://example.org/data/your_csv_file.csv#row=2>
        :ID "2.0";
        :Title "Star Wars"@en-ca;
        :Year_Of_Release_USA "1977.0".
<http://example.org/data/your_csv_file.csv#row=3>
        :ID "3.0";
        :Title "Aliens"@en-ca;
        :Year_Of_Release_USA "1986.0".
<http://example.org/data/your_csv_file.csv#row=4>
        :ID "4.0";
        :Title "Dope"@en-ca;
        :Year_Of_Release_USA "2015.0".
```

Figure 3. Dataset generated by data depositor

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix dc: <http://purl.org/dc/elements/1.1/> .
@prefix data: <http://exampleGUDR.org/myturtle.xlsx/data/Sheet 1#> .
@prefix vc: <http://exampleGUDR.org/myturtle.xlsx/voc/Sheet 1#> .

<http://exampleGUDR.org/myturtle.xlsx/voc/Sheet 1#>
        dc:title "myturtle.xlsx_Sheet 1" ;
        dc:description "dist1" .

vc:ID a rdfs:Class ;
        rdfs:label "ID" ;

vc:Title a rdfs:Class ;
        rdfs:label "Title" ;

vc:Year Of Release USA a rdfs:Class ;
        rdfs:label "Year Of Release USA" ;
```

Figure 4. Vocabulary generated by data depositor

```
 @prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>.
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>.
@prefix data: <http://exampleGUDR.org/data/myturtle.xlsx/Sheet 1#>.
@prefix xsd: <http://www.w3.org/2001/XMLSchema#>.
@prefix vc: <http://exampleGUDR.org/voc/myturtle.xlsx/Sheet 1#>.
@prefix meta: <http://exampleGUDR.org/meta/myturtle.xlsx/Sheet 1#>.
@prefix dc: <http://purl.org/dc/elements/1.1/> .
@prefix dcterms: <http://purl.org/dc/terms/> .

<http://exampleGUDR.org/meta/myturtle.xlsx/Sheet 1#>
        dcterms:issued “2017-11-04”^^xsd:date ;
        dcterms:modified “2017-11-07”^^xsd:date ;
        dcterms:title “myturtle.xlsx/Sheet_1_Metadata”@en .
```

Figure 5. Metadata generated by data depositor

second row to have data entries. The data depositor has a command line user interface, so that users can specify addition input arguments before starting the process to specify additional information such as the language of the dataset, the base IRI and additional ontologies. Figure 3 shows the generated RDF dataset. Each row of the spreadsheet is translated into a set of RDF triple statements. Specifically, each subject has 3 predicates and three objects, corresponding to the 3 attributes and their values. The object literals include a language tag to specify that the language is in Canadian English. This comes from user input argument. Figure 4 shows the vocabulary generated by the data depositor. It defines the terms to be used by the dataset, which are the attributes of the original dataset. Figure 5 shows the generated metadata, which provides additional information about the dataset.

4.2. User Profiling

To demonstrate the benefits of the user registration system, the *Ontology of Skill and Competency Management* is adapted and our solution inherit the effectiveness of the formal ontology in representing user expertise in the GUDR [25]. The ontology entails all the competency questions. See Appendix 2 for a detailed discussion of each competency question. The ontology is consistent, and hence can be used to deduce logical conclusions. Data on trust and credibility of sources can be arbitrarily large, but not all of this data is necessary for reasoning about skills and competencies of an individual. Hence, the ontology scales well in real world applications in terms of the information related to one person. A prototype decision support system was implemented using the ontology at Novator Systems to support HR decision whose objective was to match individuals to a set of job requirements and allow competency gap analysis. The decision support system shows its effectiveness by supporting the decision making process in real-world scenario.

4.3. Testing Community Setup

We performed experiments to test the reputation-based trust supporting framework by using an artificial testing community. This section describes the testing community setup. The testing community consists of N peers. In the last experiment, N is set to be a variable to evaluate the scalability of the solution. Otherwise, N is set to 128. The game theory research on reputation

introduced two types of players [33]. One is commitment type or long-run player who would always cooperate because cooperation is the action that maximizes the player's payoffs in long run if the player could reliably commit to an action throughout the entire process. The other is a strategic type or an opportunistic player who cheats upon cases that benefits him. We define two types of users in our testing community according to this standard. The percentage of malicious users, or strategic users is denoted by k. The default value of k is set to 25%.

The behaviour pattern for good users is straightforward, which is to provide reliable datasets and submit accurate and honest feedback to other user's dataset after usage. On the other hand, the malicious behaviour is non trivial. For example, a user may decide to be a good data provider but a bad rater, meaning he deposit good quality data into the repository, but often submit false statements about other user's data. In this experiment, we define malicious user as user who is a bad data provider and a bad rater. Other scenarios include being at the same time a good data provider and a good rater, a bad data provider but a good rater, and a bad data provider and a bad rater. In addition, it is not realistic for a user to behave maliciously at all time, so we define r_m to be the rate that a malicious user acts maliciously. r_m was set to 100% for the initial experiment. To simulate the effect of user verification, we introduce r_v to denote the percentage of verified users and is set to 70%. Table 3 summarizes the experiment parameters.

Experiment Parameters

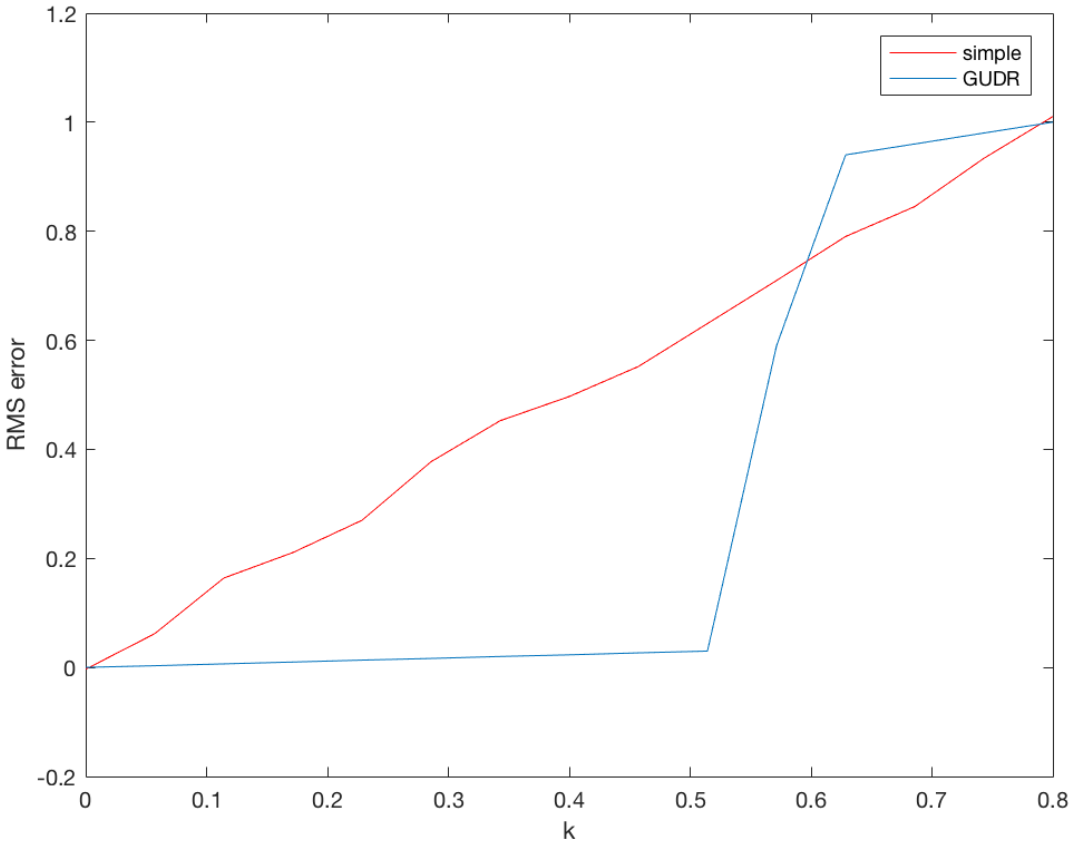
	Parameter	Description	Value
Community Setting	N	Number of users in the community	128
	k	Percentage of malicious users in the community	25
	r_m	Percentage of time a malicious user behaves maliciously	100
Trust Computation	D	Number of datasets a data provider u deposited in recent time window	100

Table 3. Experiment parameters

We uses a numeric rating system where a rater submit numeric ratings that is either 0 or 1 to indicate users' level of satisfaction regarding a dataset. The number of datasets a data provider u deposited into the repository in the last 6 months, i.e. the latest time window, is denoted by D and is set to 100 for all users.

4.4. Trust Evaluation Component

We use the experiment to evaluate the effectiveness and robustness of our trust evaluation component in the presence of malicious users. At the beginning, users randomly choose the dataset to use. Under this setting, a good user has a true trust score of 1 and a malicious user has a trust score of $(1 - r_m)$. After performing sufficient amount of updates to the trust scores of all users in the community, we use root-mean-suqre (RMS) error between the estimated trust scores and the ground truth trust scores to evaluate the performance of our system. A low RMS error is desired. To the purpose of comparison, a basic community which uses a simple summation to calculate trust scores is simulated. As we vary the percentage of malicious users in the



community, the two algorithms behave differently. It can be observed that as the number of malicious users increase in a community, the performance of the simple metric drops linearly. This indicates the vulnerability of the simple metric against malicious users. Our approach, on the other hand, stays effective when k is less than 0.5, meaning out approach is able to filter out malicious users and make the correct trust evaluation. However, the error grows quickly as k exceeds 0.5, which indicates out system makes completely wrong evaluations by recognizing good users as untrustworthy and malicious users as trustworthy. In other words, the malicious users fool the system when they are the majority. The reputation-based trust supporting system stays effective when no more than half of the community are strategic users. This is a reasonable assumption in the context of the GUDR. Hence, our solution would effectively and robustly estimate the trustworthiness users in the GUDR and provide users with an accurate trust approximation.

5. Conclusion

This thesis report documents my progress in my thesis research. The context for the research is concisely presented in section 1. To better understand the problem and its background, a literature review is included, which serves as a critical summary of published literature relevant to the evaluation of data trustworthiness for crowdsourcing data. The literature review identifies three research challenge. First, how to translate datasets of various formats into a single representation, and build explicit metadata and vocabulary. Second, how to retrieve initial identifying information from users and maintain a continuous flow of information to update an individual's expertise. Third, how to assess the quality of data in the GUDR. To address the research gap, we develop a three component solution consisting a data depositor to translate spreadsheets into RDF triples and attach metadata to datasets, a user registration system to develop user profiles in the GUDR, and a reputation-based trust supporting framework which quantifies and evaluates the trustworthiness of users based on a crowdsourcing feedback system. The trust evaluation component is responsible for computing the trust measure based on the reputation data which are collected from registration system about users. In section 3.3, we have presented four important trust parameters used by our trust model and formalized these

parameters, presented a general trust metric that combines these parameters and introduced formulas we use to compute these parameters. In the near future, our work on the GUDR depositor services will be along the following directions.

1. *Registration system:*

The current status of our registration system allows users to self declare skills, skill proficiencies, past experience, and degrees during registration. To better exploit the completeness of the skill ontology [25], it is desired to have the ability to also declare learning activity and user declared content in the system. Moreover, we plan to specialize our user registration system to differentiate between individuals users and organizations. For example, how do we automatically learn the organization hierarchies given multiple users belonging to the same organization but different departments register in the GUDR. We plan to study the performance of our framework into more details through additional experiments. The experiments should further evaluate the effectiveness of our framework against different computation overhead, and the dynamic personality of users. Besides reputation evaluation, it would add value to the dataset by providing basic metrics such as the sizes, spareness, and timeliness. Such metrics should be easy to compute and insensitive to the size of the database. We plan to design a feedback questionnaire, so users can provide feedback on these metrics to reflect the correctness of these metrics. We use the questionnaire to correct our algorithms

2. Trust Computation:

Although the trust model is independent of its implementation, the effectiveness of supporting trust in the GUDR depends heavily on the implementation. Typical issues in implementing a trust model includes secure trust data management. For example, how to efficiently and securely store and look up trust data that is needed to compute the trust scores. To answer such questions, more research need to be done.

3. Extended depositor:

Appending the depositor algorithms to incorporate other data types such as RDF/OWL and MySQL database file. Generalizing the algorithms to work with well structured data is expected to be more challenging than spreadsheets because the algorithms need to recognize the internal

structure of the input data and properly translate information encoded in input data to output data. In addition, the data depositor currently does not support mapping to existing ontologies. This is, however, valuable to the GUDR since this helps making links between related data and contribute to the Semantic Web Vision. Therefore, the next step is to recognize existing ontologies used in a dataset, and make the correct mapping to associated the ontologies to the dataset explicitly.

4. Policy Language:

My work provides a quantitative measure for the trustworthiness of both datasets and data providers. But data users still need to decide what is a good trust score. The next step is to develop a policy language to recommend what is the minimum trust score a dataset must have for users in specific roles. Such recommendation system can helps users to make decisions.

Reference

[1] P. Mooney, P. Corcoran, and A. C. Winstanley, Towards quality metrics for openstreetmap

[2] P. Sztompka, Trust: A sociological theory, Cambridge University Press, 1999.

[3] Turner, John C., Michael A. Hogg, Penelope J. Oakes, Stephen D. Reicher and Margaret S. Wetherell. 1987. Rediscovering the Social Group: A Self-Categorization Theory. New York: Basil Blackwell.

[4] Thoits, Peggy A. And Lauren K. Virshup. 1997. "Me's and We's: Forms and Dunctions of Social Identities."

[6]K.E. Seamons, M. Winslett, and T. Yu, "Limiting the Disclosure of Access Control Policies During Automated Trust Negotiation," Proc. Network and Distributed System Secu- rity Symp., IEEE CS Press, Los Alamitos, Calif., 2001, pp. 109-125.

[7] Swann, W. B., L. P. Milton, J. T. Polzer. 2000. Should we create a niche or fall in line? Identity Negotiation and small group effectiveness.

[8] Wasko, M., S. Faraj. 2005. Why should I share? Examining social capital and knowledge contribution in electronic networks of practice.

[9] Ackerman, M. S. 1998. Augmenting the organizational memory: A field study of answer garden

[10] Bock, G.-W., R. W. Zmud, Y.-G. Kim, J.-N. Lee. 2005. Behavioral intention formation in knowledge sharing: Examining the roles of extrinsic motivators, social-psychological forces, and organizational climate

[11] C. Dellarocas, "The digitization of Word of Mouth: Promise and Challenges of Online Reputation Reporting Mechanism," *Management Science,* vol. 49, no. 10, 2003

[12] P. Resnick, R. Zeckhauser, E. Friedman, and K. Kuwabara, "Reputation Systems," *Comm. ACM,* vol 43, no. 12, 2000

[13] https://www.facebook.com/help/1288173394636262

[14] https://www.facebook.com/help/100168986860974/

[15] https://help.twitter.com/en/managing-your-account/about-twitter-verified-accounts

[16] https://support.google.com/business/answer/4566671?hl=en&ref_topic=4854130

[17] L. Xiong and L. Liu. Peertrust: Supporting reputation-based trust for peer-to-peer electronic communities. *IEEE Trans. Knowl. Data Eng.,* 16(7):843-857, 2004

[18] C. Dellarocas, "Analyzing the Economic Efficiency of Ebay-Like Online Reputation Reporting Mechanisms," *Proc. Third ACM Conf. Electronic Commerce,* 2001

[19] C. Dai, D. Lin, E. Bertino, and M. Kantarcioglu, "An Approach to Evaluate Data Trustworthiness Based on Data Provenance," in SDM '08: Proceedings of the 5th VLDB workshop on Secure Data Management, pp. 82–98, 2008.

[20] Berners-Lee, T., Hendler, J., Lassila, O. (2001) The Semantic Web. Scientific American, 284(5):34-43.

[21] http://dublincore.org/documents/dcmi-terms/

[22] *W3C*, www.w3.org/standards/semanticweb/ontology.

[23] M. S. Fox, "Global Urban Data Repository", Internal Document

[24] Earl, M. (2001). "Knowledge Management Strategies: Toward a Taxonomy", in *Journal of Management Information Systems*, 18(1), 215–233

[25] M. Fazel-Zarandi, M. S. Fox, "Inferring and Validating Skills and Competencies over Time", *Applied Ontology*, *8*(3), 131–177.

[26] P. Resnick, R. Zeckhauser, E. Friedman, and K. Kuwabara, "Reputation Systems," Comm. ACM, vol. 43, no. 12, 2000.

[27] Barbier, G., Zafarani, R., Gao, H. et al. Comput Math Organ Theory (2012) 18: 257. https://doi.org/10.1007/s10588-012-9121-2

[28] Bizer, C., Cyganiak, R. (2006). D2R Server - Publishing Relational Databases on the Semantic Web. Poster at the 5th International Semantic Web Conference (ISWC2006).

[29] Knoblock C.A. et al. (2012) Semi-automatically Mapping Structured Sources into the Semantic Web. In: Simperl E., Cimiano P., Polleres A., Corcho O., Presutti V. (eds) The Semantic

Web: Research and Applications. ESWC 2012. Lecture Notes in Computer Science, vol 7295. Springer, Berlin, Heidelberg

[30] Jingwei Huang , Mark S. Fox, An ontology of trust: formal semantics and transitivity, Proceedings of the 8th international conference on Electronic commerce: The new e-commerce: innovations for conquering current barriers, obstacles and limitations to conducting successful business on the internet, August 13-16, 2006, Fredericton, New Brunswick, Canada [doi>10.1145/1151454.1151499]

[31] C. Dellarocas, "Analyzing the Economic Efficiency of Ebay-Like Online Reputation Reporting Mechanisms," Proc. Third ACM Conf. Electronic Commerce, 2001.

[32] Bizer, C.: Quality-Driven Information Filtering in the Context of Web-Based Information Systems. PhD thesis, Freie Universität Berlin (2007)

[33] C. Dellarocas, "The Digitization of Word-of-Mouth: Promise and Challenges of Online Reputation Mechanism," Management Science, vol. 49, no. 10, 2003.

11.1. Skill Measurement Module

Predicate	Informal Definition
set(X)	Sets are objects with at least one member.
interval(X)	Intervals are objects with min and max values.
measured-attribute(M)	A measurable attribute related to a skill.
measurement-unit(U)	Unit of measurement for a measured-attribute.
specification-set(Sp)	A set of values which denotes possible values for a measured- attribute.
prof iciency-level(L)	Refers to the ranking of the ability of an individual to perform the activities enabled by a particular skill.
set-member(X, Y)	Object *X* is a member of set *Y*.
in-interval(X, Y)	*X* is in interval *Y*.
has-spec(M, Sp)	Measured-attribute M has specification set Sp.
has-unit(M,U)	Measured-attribute M has unit of measurement U . .
sless-than(X, Y)	This relation is used to impose an ordering on members of a set.
lesser(X, Y)	This relation is used to impose an ordering on an interval.
dominates(L_1, L_2)	Proficiency level L_1 dominates level L_2.

11.2. Skill Core Module

Predicate	Informal Definition
skill(S)	A class or type of skill. A skill suggests the possibility of per- forming an activity.
knowledge-F ield(F)	A field of knowledge.
enables(S, A, L)	Skill *S* enables activity *A* at level of proficiency *L*.

enabling-suite(S, A, L)	A complex activity that includes all the activities enabled by a skill at a particular level of proficiency.
in-f ield(S, F)	Skill *S* is in knowledge-field *F*.
related-to(S_1, S_2)	Two skills are related if they enable the same activity, or if they enable different subactivities of the same activity.
requires-value(S, L, M, X)	Skill *S* at level of proficiency *L* for measured attribute *M* re- quires value *X*.
subskill-of(S_1, S_2)	This relation indicates skill specialization, forming a taxon- omy of skills.

11.3. Organization and Trust Ontologies Modification

Predicate	Informal Definition
requires-skill(R, S, L)	Role R requires skill S at level of proficiency L.
remmends(X, Y)	Organization-agent X recommends organization-agent Y .
credible-source-f or(X, S, T)	Source X is a credible source of information for skill S at timepoint T .

11.4. Skill Statement Module

Predicate	Informal Definition
demonstrated(skill-statement(p, s, l))	Relational fluent. *p* has demonstrated skill *s* at level of pro- ficiency at least *l*.
probable(skill-statement(p, s, l))	Relational fluent. It is highly probable that *p* has skill *s* at level of proficiency at least *l*.
possible(skill-statement(p, s, l))	Relational fluent. It is possible that *p* has skill *s* at level of proficiency at least *l*.

refuted(skill-statement(p, s, l))	Relational fluent. *skill-statement(p, s, l)* has been refuted.
asserted(skill-statement(p, s, l))	Relational fluent. *skill-statement(p, s, l)* is *demonstrated*, *probable*, or *possible*.
reversible(skill-statement(p, s, l))	Relational fluent. *skill-statement(p, s, l)* is *reversible* if its state has changed to *refuted* using information other than direct observation.

11.5. Sources of Information Module

Predicate	Informal Definition
supports(X, St, O)	X supports skill statement St with evidence O.
rejects(X, St, O)	X rejects skill statement St with evidence O.
perf orms(P, O)	P performs activity-occurrence O.
declares(X, P, S, L)	Activity. Agent X declares that agent P has skill S at level L.
declares-neg(X, P, S, L)	Activity. Agent X declares that agent P does not have skill S at level L.
learning-activity(A)	Activity. An activity which has at least one skill at a level of proficiency as outcome.
has-outcome(A, S, L)	The outcome of a learning activity.
has-precondition(A, S, L)	Preconditions of a learning activity, these are skills which an individual should have in order to take the activity.
degree(D)	An object that requires a set of formal learning activities.
requires-f la(D, A)	D requires formal learning activity A.
has-degree(P, D, C)	P has degree D from educational institution C.
adds-experience(P, R, C)	Activity. Agent P has played role R at organization C.

content(X)	An object with an individual either as a creator or a contributor associated with it.
content-type(X)	A class or type of content.
activity-outcome(X, A)	Content-type X is the outcome of activity A.
end-result(X, O)	Content X is the end result of activity-occurrence O.
tags(X, Y, S)	Activity. X tags content Y with skill S.
test(X)	An object which measures the level of proficiency in at least one skill.
measures-skill(X, S, L)	Test X measures skill S at level of proficiency L.
takes-test(P, X)	Activity. Agent P takes test X.
passes(P, X, T)	P passed test X at timepoint T .
fails(P,X,T)	P failed test X at timepoint T .

Appendix 2 - Entailment of the Competency Questions

In this section, we repeat the competency questions presented in Section 3.2 and discuss how the ontol- ogy represents and answers these questions.

- Q-1 What skills are needed to perform the required activities? Given activity *A*, this question can be formally represented as: ∃s enables(s, A, l).

- Q-2 Are two skills related?
  Given skills S_1 and S_2, this question can be formally represented as: related-to(S_1,S_2)∨subskill-of(S_1,S_2).

- Q-3 What are the proficiency reference levels for evaluation against a skill? Given skill *S*, this question can be formally represented as: ∃l enables(S, a, l).

Q-4
Whatarethecriteriafordeterminingwhetheranindividualpossessesaskillatalevelof proficiency?

Given skill *S*, level of proficiency *L*, measured-attribute *M*, and measurement unit *U*, the following questions can be formally represented.

Q-4.1 What are the activities that the individual should be able to perform?

∃a enables(S, a, L).
Q-4.2 What are the attributes related to that skill that can be measured?

∃m requires-value(S, L, m, x).
Q-4.3 What is the unit of measurement for an attribute related to a skill?

∃u requires-value(S, L, m, x) ∧ has-unit(m, u).
Q-4.4 What ought to be the measured value to be ranked at a level of proficiency?

∃x requires-value(S, L, M, x).

- Q-5 What evidence suggests that an individual has a skill at a level of proficiency?
  Given individual *P*, skill *S*, and level of proficiency *P*, this question can be formally represented as: ∃o achieved(asserted(skill-statement(P, S, L)), o)∧(participates-in(P, o)∨perf orms(P, o)).
  If a skill statement is asserted, or, in other words, in any of the states *demonstrated*, *probable*, or *possible*, then a source supporting it is suggesting that the individual has the skill. We consider ev- idences as occurrences of activities. The evidence can be suggested by the performance of activi- ties enabled by a skill using axioms S-4 and S-5, or by different sources of skill and competency information using one of the axioms R-1, R-2, R-4, R-6-R-8, or R-10.

- Q-6 Which source is providing this evidence? Is it a credible source of information?

Given individual *P*, skill *S*, and level of proficiency *P*, this question can be formally represented as:
∃x supports(x, skill-statement(P, S, L), o) ∧ (participates-in(P, o) ∨ perf orms(P, o))∧ credible-source-f or(x, S, endof (o)).

- Q-7 What are the skill statements about an individual at a given point in time?
  Given individual *P* and timepoint *T*, the following questions can be formally represented.

Q-7.1 What are the *demonstrated* skills of an individual at a given time point?

∃s holds(demonstrated(skill-statement(P,s,l)),o)∧beforeEq(endof(o),T)∧ ¬(∃o′holds(refuted(skill-statement(P,s,l)),o′)∧beforeEq(endof(o),endof(o′))∧

beforeEq(endof(o′),T)).

The state of a skill statement can change to *demonstrated* only if the individual performs all the activities enabled by a skill and satisfies all the measured attributes as specified by S-4. However, if a person fails a measured-attribute later on due to knowledge decay, the state can change to *refuted* using S-6. All the other axioms will not change the state of a *demonstrated* skill statement.

Q-7.2 What are the suggested skills of an individual at a given time point, i.e. those skills that have not been observed but the individual may possess?

∃s (holds(probable(skill-statement(P, s, l)), o) ∨ holds(possible(skill-statement(P, s, l)), o)) ∧beforeEq(endof(o),T)∧¬(∃o′(holds(refuted(skill-statement(P,s,l)),o′)∨

holds(demonstrated(skill-statement(P, s, l)), o′))∧
beforeEq(endof(o),endof(o′))∧beforeEq(endof(o′),T)).

The state of a skill statement can change to *probable* if the individual has performed all the enabled activities but has not been measured yet using S-5, or using R-1, R-6-R-7. Similarly, the state of a skill statement can change to

possible using R-2, R-4-R-7. However, if a person performs all the activities enabled by a skill and satisfies all the measured attributes later on, then the state change

to *demonstrated* using S-4. Additionally, axioms R-3, R-5, and R-9 change the state of a *probable* and/or *possible* skill statement to *refuted*. All the other axioms will not change the state of a *probable* or *possible* skill statement.

Q-7.3 What are the *refuted* skills of an individual at a given time point, i.e. those skills that the individual does not have?

$\exists s\ holds(refuted(skill\text{-}statement(P,s,l)),o) \wedge beforeEq(endof(o),T) \wedge \neg(\exists o' holds(asserted(skill\text{-}statement(P,s,l)),o') \wedge beforeEq(endof(o),endof(o')) \wedge beforeEq(endof(o'),T))$.

The state of a skill statement can change to *refuted* if the individual fails at least on measured- attribute as specified by S-6 or through negative assertions by a source of skill and competency in- formation using one of the axioms R-3, R-5, or R-9. However, if a person performs all the activities enabled by a skill and satisfies all the measured-attributes later on, then the state change to *demon- strated* using S-4. Additionally, axioms R-2, R-6-R-8 change the state of a *refuted* skill statement to *probable* or *possible*. All the other axioms will not change the state of a *refuted* skill statement.

Q-8 How did belief in a skill statement change over time?

Given individual *P*, skill *S*, and level of proficiency *P*, this question can be formally represented as:

$\exists o, t\ (holds(demonstrated(skill\text{-}statement(P, S, L)), o) \vee holds(probable(skill\text{-}statement(P, S, L)), o) \vee holds(possible(skill\text{-}statement(P, S, L)), o) \vee holds(ref\ uted(skill\text{-}statement(P, S, L)), o)) \wedge endof\ (o) = t.$

The history of how belief in a skill statement has changed over time can be retrieved by querying the knowledgebase for the occurrences which affected a particular skill statement along with the ending timepoint of each occurrence.