Leveraging Matching Dependencies for Guided User Feedback in Linked Data Applications

Umair ul Hassan
Digital Enterprise Research Institute,
National University of Ireland, Galway, Ireland
umair.ul.hassan@deri.org

Sean O’Riain
Digital Enterprise Research Institute,
National University of Ireland, Galway, Ireland
sean.oriain@deri.org

Edward Curry
Digital Enterprise Research Institute,
National University of Ireland, Galway, Ireland
ed.curry@deri.org

ABSTRACT

This paper presents a new approach for managing integration quality and user feedback, for entity consolidation, within applications consuming Linked Open Data. The quality of a dataspace containing multiple linked datasets is defined in term of a utility measure, based on domain specific matching dependencies. Furthermore, the user is involved in the consolidation process through soliciting feedback about identity resolution links, where each candidate link is ranked according to its benefit to the dataspace; calculated by approximating the improvement in the utility of dataspace utility. The approach evaluated on real world and synthetic datasets demonstrates the effectiveness of utility measure; through dataspace integration quality improvement that requires less overall user feedback iterations.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval – *Relevance feedback*

General Terms

Measurement, Experimentation

Keywords

Linked data, identity resolution, user feedback, matching dependencies

1. INTRODUCTION

Linked Open Data (LOD) facilitated publishing of large amounts of structured data, that enables the creation of a global dataspace on the web [1]. As linked data becomes main stream, more web applications will increasingly consume LOD in interesting and innovative ways [2]. However, applications grapple with data quality issues due to heterogeneity of interlinked datasets [3]. A major concern relating to Linked Data quality is the problem of *identity resolution*. Due to the open nature of the LOD publication process, same real world entities are often represented with different identifiers (i.e. URIs) resulting in data publishers and data consumers having to share the burden of identity resolution [1]. This distribution of efforts introduces an uncertainty relating to the identity resolution links produced by different mechanisms. Consequently, the applications consuming Linked Open Data require further verification of identity resolution links through *user feedback*.

Human verification of identity resolution links is relatively easy for small datasets but becomes infeasible as the size and heterogeneity of the dataspace becomes sufficiently large. Therefore effective use of human attention necessitates a per link utility measurement. In this regard, *decision theoretic approaches* [4] have been proven suitable for ranking human verification tasks in problem areas such as image labeling [5], relational database repairs [6] and feedback frameworks for dataspaces [7]. Decision theoretic approaches rank tasks based on application specific *utility measures*, such as uncertainty of learning models [5], loss of data quality [6] and quality of query results [7]. In this paper, we argue that matching dependencies can be leveraged to define the utility of identity resolution links with in the dataspace, for the purpose of guiding user feedback.

Motivating Example

Consider the following example of movie data collected from three sources that represents a small dataspace:

```
<src1:movie1, imdb:Genre, “Drama”>
<src1:movie1, imdb:Genre, “Short”>
<src1:movie1, imdb:HasTitle, “Scarface”>
<src1:movie1, imdb:Year, “1928”>

<src2:movie2, movie:genre, “Short”>
<src2:movie2, rdfs:label, “Scarface”>
<src2:movie2, time:year, “1928”>

<src3:item1, cd:type, “DVD”>
<src3:item1, cd:title, “Scarface”>
<src3:item1, cd:releaseYear, “2005”>
<src3:item1, cd:price, “$3.50”>
```

In this example data is formatted using to *Resource Description Framework* (RDF) triples. A triple consists of three elements <*entity*, *attribute*, *value*>, where *entity* and *attribute* are URIs, and *value* is a string literal. In the rest of the paper we will use terms *attribute* and *property* interchangeably.

Identity resolution links are most commonly represented with *owl:sameAs* property, where each triple represents equivalence relationship between two entities. For example

```
<movie1, owl:sameAs, movie2>
```

Identity resolution links between entities may already be available with data publishers. Otherwise, an application can utilize existing matching tools[1] to generate candidate links. In any case, there is an uncertainty associated with the links, either due to the specifics

IIWeb’12, May 20 2012, Scottsdale, Arizona, USA.
Copyright 2012 ACM 978-1-4503-1239-4/12/05…$10.00.

[1] www.instancematching.org

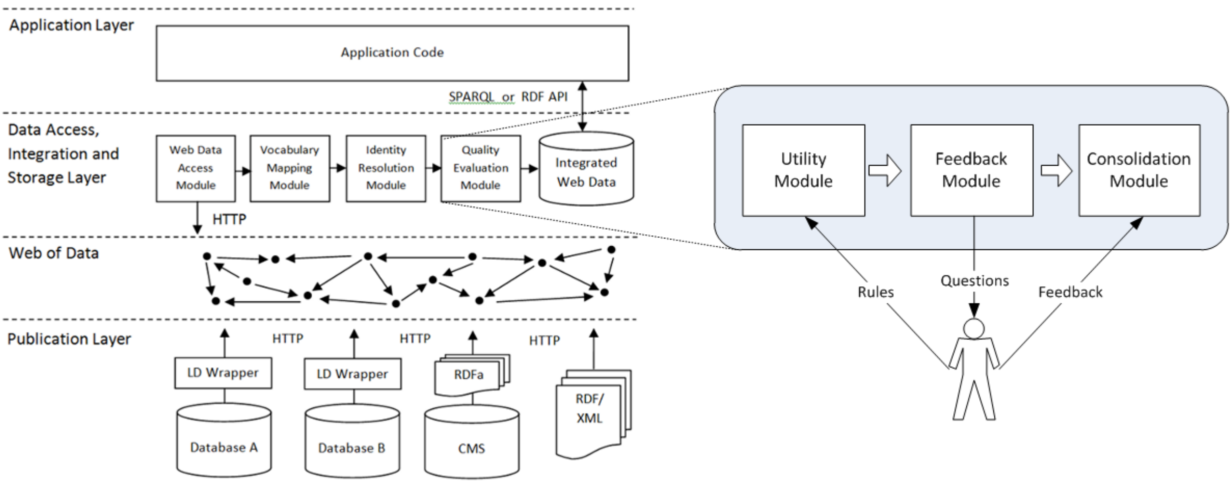

Figure 1: Architecture of entity consolidation with rules and user feedback in a Linked data application [1]

of the integration algorithm or the data source. This uncertainty requires human verification usually performed by asking questions, of the user for each identification link, soliciting binary responses.

Problem Definition

Considering that the applications consuming LOD have to deal with large quantity of data and identity resolution links, it becomes infeasible to verify all of the uncertain links before consolidating entities. Alternatively, links can be ranked for user feedback. A naive approach can consider the number of RDF triples (associated with a link) as a basic ranking mechanism. Sophisticated approaches however define utility of dataspace based on query results quality [7]. Query based approaches assume the availability of global query processing information about a dataspace, such as statistics relating to elements and distribution of query workloads. Since this assumption is unrealistic for LOD on the Web, our goal is to define utility in terms of domain specific constraints, which are relevant to the application's datasets.

Data dependencies [8] are attracting renewed interest as effective formalisms for specifying semantics of data quality. Specifically, *matching dependencies* (*MDs*) [8], [9] define constraints on a pair of entities by specifying similarity and matching comparisons of attributes. For example, a matching dependency for the above example would be

$$\varphi_1: [rdfs: label \approx imdb: HasTitle] \rightarrow \ [time: year \rightleftharpoons imdb: Year])$$

where φ_1 specifies that for an entity pair (e_1, e_2), if the value of property *rdf:label* of entity e_1 is similar to value of property *imdb:HasTitle* of entity e_2, then the value of property *time:year* of e_1 should be matched with value of property *imdb:Year* of e_2. In this paper, we employ matching dependencies to define domain specific rules for identifiers and attribute values, in a linked dataspace.

Contributions:

In this paper, we present a utility driven approach for verifying identity resolutions links from users. Our main contributions are as follows:

- Leveraging matching dependencies for consolidation of entities in applications consuming Linked Data; in addition to definition of a utility measure for linked dataspace based on matching dependencies
- A strategy for ranking identity resolution links based on approximated utility of expected user feedback
- Experimental evaluation of the proposed approach on real world and synthetic datasets

The remainder of this paper is organized as follows. Section 2 provides an overview of the proposed approach, in the context of Linked Data applications. Sections 3, 4, and 5 describe individual system modules. In Section 6 results of an experimental evaluation are presented. Section 7 discusses some of the related research efforts, followed by concluding remarks and directions for future research in Section 8.

2. OVERVIEW

Figure 1 details the system architecture within the context of Linked Data application [1]. Applications follow a three stage process that first collects data by traversing RDF links; then cleans and integrated the data, before presenting a high quality consolidated view. The consolidation process starts after web data access, vocabulary mapping, and identity resolution have been performed. We assume that the user specifies domain specific quality rules, or uses existing tools [9] to infer rules from data. These rules represent user requirements of data quality with in the entity consolidation process. The three main modules of consolidation process are as follows

- The *utility module* maintains a list of domain specific rules and calculates the utility of a dataspace as well as individual links.
- The *feedback module* calculates the ranking based on the expected benefit of verifying identity resolution links. Additionally, it generates questions for each link along with the necessary data to support user's decision.
- The entity *consolidation module* utilizes user feedback to merge data of identical entities to produce high quality integrated web data.

We specifically focus on the utility and feedback modules, opting to leave the discussion on consolidation module for future work. The rest of the paper presents the application of matching dependencies for dataspace utility calculation and ranking of identity resolution links.

2.1 Identity Resolution

Identity resolution (also known as entity resolution [1], duplicate detection [10], and instance matching [11]) is an essential part of any web data integration process, which involves finding equivalence relationships between different identifiers of same real world objects. A Linked Data application can collect identity resolution links using three different methods

- Links provided by data publishers. For example *dbpedia.org* provides links to *freebase.com* and *linkedmdb.com*
- Links generated by using automated tools or libraries such as SILK, LIMES, SEMIRI, RiMOM, etc.
- Links maintained and published by third parties such as *okkam.org* and *sameas.org*

The identity resolution links for the entities discussed in the earlier example would be

$$m_1 = (< movie1, owl{:}\, sameAs, movie2 >)$$

$$m_2 = (< movie1, owl{:}\, sameAs, movie3 > ,0.89)$$

The problem of automated identity resolution for Linked Data has attracted significant amounts of research proposals in recent years [10], [11]. Approaches range from rules-based to sophisticated machine learning techniques. Rules-based approaches require significant upfront domain knowledge and manual parameter tuning to generate links between entities. Automated learning based approaches on the other hand suffer from accuracy issues.

The majority of matching algorithms produce a similarity score that correlates with the confidence of match between two entities; however this score can be an inaccurate representation of the probability of correctness. Therefore, the probabilities of match can be approximated using histograms, as discussed in [7]. Furthermore thresholds can be applied to filter false positives from potential links.

Due to the open nature of LOD, there is an inherent uncertainty associated with identity resolution links. The situation is further exasperated by their relevance to the semantics of the particular application domain. Semi-automated tools (such as SILK and LIMES) allow control over the matching process by allowing specification of detailed transformations and similarity comparisons. However, the process of defining specification is tedious, time consuming, and still requires manual verification of output links to establish quality of matching.

Our research assumes that the utility module has access to links collected through either of the above mentioned methods. Additionally for the case of link generation tools, either the application is agnostic to the semantics of matching algorithm or relatively little effort (in terms of linkage specifications and thresholds selection) is spent for list generation of potential candidate links. The goal of this paper is to 1) develop matching dependencies based approach that allows systematic measurement of the utility of entity pairs and 2) rank identity resolution links according to approximated utility where user feedback is not known beforehand.

2.2 Matching Dependencies

Matching dependencies (MDs) [9] define constraints over a pair of entities with defined dynamic semantics in terms of *similarity operators* (possibly across *different schemas*) to cope with errors in attribute values. Consider a dataspace D of RDF triples describing entities over multiple data sources. Let $e \in E$ be an entity, $a \in P$ be an attribute, $e[a]$ be value of attribute, and $< e, a, e[a] >$ be a triple in D. A matching dependency φ, for an entity pair (e_1, e_2) is syntactically represented

$$\bigwedge_{a\in P}(e_1[a_1] \approx e_2[a_2]) \to \bigwedge_{a\in P}(e_1[a_3] \rightleftharpoons e_2[a_4])$$

where $\approx$ denotes a similarity operator and $\rightleftharpoons$ denotes a matching operator. The *similarity operator* (between two compatible attributes) states the general or domain specific similarity metrics such as edit distance, cosine similarity, etc. The semantics of *matching operator* state that the values of attributes should be treated as equal. To further explain, the following SPARQL[2] query returns a list of entity pairs satisfying φ_1 discussed earlier in the movies example

```
SELECT ?e1 ?e2
WHERE {
        ?e1 rdfs:label ?label .
        ? e1 time:year ?year .
        ?e2 imdb:HasTitle ?HasTitle .
        ?e2 imdb:Year ?Year .
        FILTER MATCH(?label, ?title) .
        FILTER (?year<>?Year) }
```

where *MATCH* is a user-defined function that returns a Boolean value according to similarity between arguments. A special case of MDs defines matching rule for identity resolution. In this case the semantics of matching entities (i.e. their respective identifiers) are based on similarity of values of their attributes. For example

$$\varphi_2{:}\, [rdfs{:}\, label \approx imdb{:}\, HasTitle] \to\ [e_1 \rightleftharpoons e_2])$$

The corresponding SPARQL for φ_2 would be

```
SELECT ?e1 ?e2
WHERE {
        ?e1 rdfs:label ?label .
        ?e2 cd:title ?title .
        FILTER MATCH(?label, ?title) .
        FILTER (?e1=?e2) }
```

The utility of a matching dependency is defined according to validation of its satisfaction in the dataspace. Given a pair of entities $z_{ij} = (e_i, e_j)$ and matching rule φ, the satisfaction function denoted as $sat(z_{ij}, \varphi)$, is defined as

$$sat(z_{ij}, \varphi) = \begin{cases} 1 & if\ (e_i, e_j)\ satisfies\ \varphi \\ 0 & otherwise \end{cases}$$

Following this definition, we define utility of a matching rule φ denoted as $|D \vDash \varphi|$ as

$$|D \vDash \varphi| = \frac{\sum_{z_{ij}\in D} sat(z_{ij}, \varphi)}{|E|^2}$$

where, $|E|$ is the number of distinct entities in the dataspace. Since dependencies provide a useful formalism for detection and repair of inconsistencies in data. We will later discuss how adapting dependencies is beneficial for effectively managing quality in Linked Data applications. The next section discusses quality of a linked dataspace in terms of matching dependencies and identity resolution links.

[2] Query language for RDF datasets and databases

3. UTILITY MODULE

Dataspace utility quantifies the quality of integration from a user and application perspective. Using pre-defined rules, we consider the utility function proportional to the degree of rules satisfaction in dataspace. Let us assume, given a set of candidate identity resolution links $M = \{m_1, \dots, m_k\}$ collected or generated in a dataspace D. To define utility $U(D, M)$, measure of quality with respect to rule $\varphi \in \Phi$ needs to be defined, where Φ is set of all data quality rules. Suppose that the utility of a perfect dataspace D^P is already known, then the quality of current dataspace w.r.t φ can be denoted by:

$$q(\varphi, D, M) = \frac{|D \models \varphi|}{|D^P \models \varphi|} \tag{1}$$

where $|D \models \varphi|$ and $|D^P \models \varphi|$ is the satisfaction measures of rule φ in current dataspace D and perfect dataspace D^P, respectively. Subsequently the utility of dataspace over set of rules Φ is defined as

$$U(D, M) = \sum_{\varphi_i \in \Phi} q(\varphi_i, D, M) w_i \tag{2}$$

where w_i is weightage of rule φ_i, which can be manually tuned by user or based on ratio of entities relevant to the rule.

Within the context of user feedback for identity resolution, the user can either confirm or reject a candidate link m_k. We can denote the two states of dataspace after user feedback as $D^+_{m_k}$ and $D^-_{m_k}$, respectively. Further assuming that the probability of the update m_k being correct is p_k. Since perfect dataspace D^P is unknown, approximations are made to calculate estimated utility $EU(D, M)$. The first approximation considers $M = \{m_k\}$, where m_k is candidate link with confidence c_k. Secondly, Equation 1 becomes weighted sum of the above mentioned two possibilities, where each possibility is approximated by its respective confidence. Following this expected quality with respect to rule φ becomes

$$\begin{aligned} E[q(\varphi, D, \{m_k\})] = {} & \frac{|D \models \varphi|}{|D^+_{m_k} \models \varphi|} c_k + \\ & \frac{|D \models \varphi|}{|D^-_{m_k} \models \varphi|} (1 - c_k) \end{aligned} \tag{3}$$

Now rewriting Equation 2 for expected utility of dataspace with respect to $M = \{m_k\}$

$$\begin{aligned} EU(D, \{m_k\}) = {} & \sum_{\varphi_i \in \Phi} w_i \frac{|D \models \varphi_i|}{|D^+_{m_k} \models \varphi_i|} c_k + \\ & \sum_{\varphi_i \in \Phi} w_i \frac{|D \models \varphi_i|}{|D^-_{m_k} \models \varphi_i|} (1 - c_k) \end{aligned} \tag{4}$$

4. FEEDBACK MODULE

Value of perfect information (VPI) quantifies information value for a decision problem; to the extent that one plan is considered better than another plan. We apply the same technique for approximating benefit of identity resolution links, according to the extent of utility improvement associated with it.

Therefore, the expected gain in utility of a dataspace after user has provided feedback on a candidate link m_k can be expressed as:

$$\begin{aligned} g(m_k) = {} & U(D^+_{m_k}, M - \{m_k\}) p_k + \\ & U(D^-_{m_k}, M - \{m_k\})(1 - p_k) - \\ & U(D, M) \end{aligned} \tag{5}$$

Table 1: Summary of the main characteristics of the datasets used for experimental evaluation.

Characteristic	IIMB	UCI-Adult	Drug
Total Triples	291	64000	14348
Total Entities	44	4000	5696
Total Attributes	9	16	3
Total Values	130	10878	8473
Candidate Links	81	72	94
Correct Links	22	72	66

Here candidate links are assumed to be independent from each other, that is to say that the sequence of update confirmation does not affect eventual improvement in utility of the dataspace.

Given expected utility, we reformulate Equation 5 by substituting $U(D, M)$ with $EU(D, \{m_k\})$ by considering $M = \{m_k\}$. The expected gain in utility of the dataspace after feedback can be expressed as

$$\begin{aligned} E[g(m_k)] = {} & EU(D^+_{m_k}, \{\}) c_k + \\ & EU(D^-_{m_k}, \{\})(1 - c_k) - \\ & EU(D, \{m_k\}) \end{aligned} \tag{6}$$

Since $E[q(\varphi, D^+_{m_k}, \{\})]$ and $E[q(\varphi, D^-_{m_k}, \{\})]$ both evaluate to 1 and also the second term in Equation 4, simply rearrangement Equation 6 becomes

$$\begin{aligned} E[g(m_k)] &= \sum_{\varphi_i \in \Phi} w_i c_k - \sum_{\varphi_i \in \Phi} w_i \frac{|D \models \varphi_i|}{|D^+_{m_k} \models \varphi_i|} c_k \\ &= c_k \sum_{\varphi_i \in \Phi} w_i \left(1 - \frac{|D \models \varphi_i|}{|D^+_{m_k} \models \varphi_i|} \right) \end{aligned} \tag{7}$$

As a further optimization the number of rules in Φ can be limited to only those which are relevant to entities in the current link.

5. CONSOLIDATION MODULE

The final step of the process is to incorporate user feedback in the entity consolidation process. Entities with confirmed identity resolution are merged to create a clean and consistent view of data. Advance data fusion techniques [12] can be applied to resolve inconsistencies of attribute values. An in-depth discussion on this step is out of scope for this paper. It should be noted that the overall process of quality assessment and user feedback can executed iteratively, to support a *pay-as-you-go* approach for quality improvement.

6. EXPERIMENTS

This section details experiments on synthetic and real world datasets to evaluate the proposed utility calculation and feedback ranking approach. The experiments were performed on 2.5 GHz machine with 4 GB memory.

6.1 Datasets

Table 1 summarizes three datasets used for evaluation, denoted as IIMB-2009, UCI-Adult and Drug.

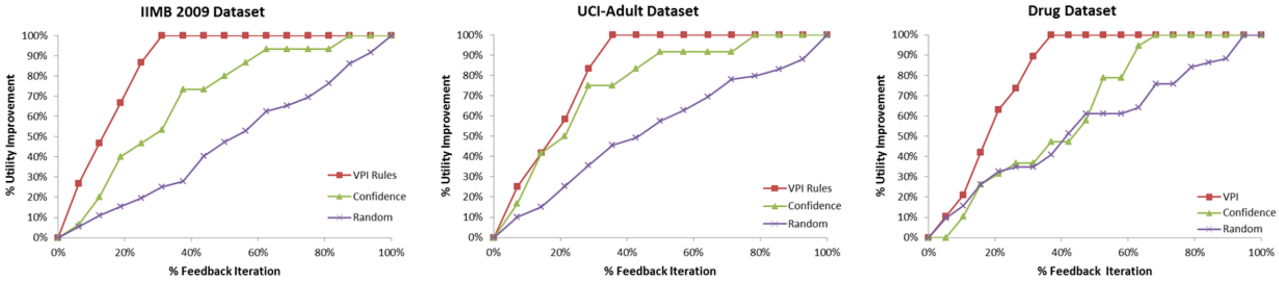

Figure 2: Comparison of user feedback ordering strategies for IIMB-2009, UCI-Adult and Drug datasets. The graphs show incremental increase from start where no feedback is available to end where perfect dataspace with all feedback is achieved.

IIMB-2009 Dataset

Our first dataset was based on Instance Matching Benchmark 2009[3]. Data about entities representing movies was selected. The dataset contains copies of movie entities with systematically introduces errors such as omission of attributes and their values according to various parameters like character changes, attribute deletion and value removals. The reference entity matches between the original and its copies are also available with the source dataset working as baseline.

A dataspace was created by integrating the original IIMB-2009 dataset with one of its copies. Identity resolution links were generated by using the SILK Framework[4]. Similarity between pairs of movies was calculated using Jaro-Winkler[5] edit distance on *HasTitle* attribute. Matching dependencies were created manually by defining thresholds for entity matches based on thresholds for *HasDirector* and *Year* attributes.

UCI-Adult Dataset

The UCI-Adult[6] dataset available online with UCI Machine Learning repository was used for generating a derived dataset with required properties. Since the dataset contains data of anonymous persons, identity for records was created by choosing names randomly from names in US Census 1990[7].

We manually created 1,000 duplicates for sample of 3,000. Additionally, values of randomly selected attributes where changed until 20% of total entities were dirty. Matching dependencies were manually created by defining similarity metrics and thresholds for attributes *firstname* and *lastname*. Table 1 describes features of the UCI-Adult dataset used for evaluation. Candidate identity resolution links were generated using SILK framework.

Drug Dataset

The Drug dataset is based on data interlinking track of the Instance Matching Benchmark 2010[8]. This dataset contains data entities from the biomedical domain such as drugs, diseases, companies, etc. The reference alignment between entities of different sources is also available with the original datasets working as a baseline. For the purpose of evaluation DrugBank and SIDER[9] datasets were merged to create an integrated dataset. Feedback candidates links were generated by using the SILK framework for matching small number of drug entities. The similarity between drugs was calculated using Jaro-Winkler edit distance on *rdf:label* attribute. Matching dependencies for the datasets were created manually by defining thresholds for entity matches based on thresholds for *rdf:label* and *drugbank:genericName* attributes.

6.2 Evaluation Settings

Evaluation of the proposed approach is based on the improvement in utility defined in terms of dependencies. The experiment objective is to demonstrate that the utility function effectively helps in ranking candidate links.

Utility Measurement: Dataspace utility (detailed in Section 3) is calculated using simulated feedback form the manually created gold standard for each dataset. After feedback confirmation for every fifth candidate link, utility of the complete dataspace is recorded. We measure the percentage improvement in utility from the start dataspace instance D_0 and then for each iteration till the last recorded utility considering it as perfect dataspace D_P. Original baseline dataset and reference entity alignments were used to confirm the user feedback. After each confirmation, the relevant link is then added to the data store.

Ranking Strategies: The following feedback ranking strategies have been evaluated for improvement in utility

- **Random**: Assigns a random weight to each feedback candidate according to uniform distribution between 0 and 1, for ranking.
- **Confidence**: Considers the confidence score generated by the identity resolution algorithm as ranking criteria.
- **VPI-Rules**: Calculates ranking weight according to approximated feedback benefit (see Equation 7) to dataspace in terms of utility.

Rule Importance: The utility function also considers the importance of rules on user assigned weights. A weighting factor that is based on the number of entities or entity pairs can be considered as proxy for user assigned weights. For evaluation purpose each rule is assigned equal weight i.e. $w_i = 1$.

6.3 Results

The results of ranking strategies for all three datasets are reported. The utility of the dataspace was measured for each ranking strategy 5 times with the same dataset, candidate links and matching rules. The reported results are based on average utility of the dataspace after feedback iteration.

As illustrated in Figure 2, the *VPI-Rules* strategy performs better on both the dataset for ranking identity resolution links. The improvement in dataspace utility for *VPI-Rules* reaches its

[3] http://islab.dico.unimi.it/content/iimb2009/
[4] http://www4.wiwiss.fu-berlin.de/bizer/silk/
[5] http://en.wikipedia.org/wiki/Jaro–Winkler_distance
[6] http://archive.ics.uci.edu/ml/datasets/Adult
[7] http://www.census.gov/genealogy/www/data/index.html
[8] http://oaei.ontologymatching.org/2010/im/index.html
[9] http://www4.wiwiss.fu-berlin.de/sider/

maximum after only 35% of the links have been confirmed. In contrast, a simple strategy like *Confidence* requires feedback for between 70%-80% of candidate links before reaching maximum utility. Note that in case of IIMB-2009 dataset the difference of improvement between *VPI-Rules* and *Confidence* strategies is much higher. This is due to the fact that matching rules were defined for attributes different from the attributes used for generating candidate links. This validates our approach which directs feedback towards candidates based on the quality of dataspace defined in terms of user defined rules.

7. RELATED WORK

Research for assessment of data quality for Linked Data applications is still in its infancy. WIQA (Web Information Quality Assessment) [13] is a framework for defining policies for filtering low quality Linked Data. Hartig [14] extended syntax of SPARQL by proposing new operators for enabling trust aware query processing. Fürber et al. [15] have proposed a SPIN (SPARQL Inference Notation) based approach for identifying data quality problem in Linked Data, additionally they have define a comprehensive vocabulary for representation of various aspects of Linked Data quality. In contrast, this paper takes a quantitative metrics based approach for managing quality and guiding user feedback in support of the consolidation process with Linked Data applications.

Leveraging user's attention for improving semantic integration in dataspaces [16], is considered an integral part of any dataspace application or platform. Roomba [7] is one of the initial approaches that exploits user feedback for improving integration of dataspaces. Our approach employs a similar decision-theoretic technique to quantify desirability of a dataspace state. However, Roomba defines utility of the dataspace in terms of quality of results of queries over dataspace. This measure of quality based on cardinality of triples in a keyword query result is a query-centric measure of utility. In comparison, our approach defines utility in terms of domain specific rules that capture dynamic semantics of data. Moreover, Roomba requires global information about dataspace, such as existing query work load and element statistics. By contrast, our approach overcomes this requirement by defining utility in terms of matching rules.

Another decision theoretic approach was proposed by Yakout el at. [6] for repairing relational database tables. They use *conditional functional dependencies* (CFDs) as constrains for a relational table and solicit user feedback for attribute value repairs. In this case, the utility is based on violations of CFDs in a single relational table. Our approach adapts matching dependencies for entity consolidation over multiple linked datasets.

8. CONCLUSION

This paper presents a framework for combining matching rules and user feedback for improving the quality of consolidation in linked dataspaces. The proposed strategy (*VPI-Rules*) ranks uncertain identity resolution links according to their potential benefit to the dataspace. The utility is quantified in terms of matching dependencies which serve as domain specific constraints over entity pairs. Experimental results have shown that this approach indeed improves integration quality with fewer iterations of user feedback.

This paper presents our preliminary work on a systematic study of utility and user feedback within Linked Data applications. Future work includes extending the proposed approach with other types of data quality constraints such as comparable and order dependencies. Research into multi-user feedback is another open area as well as feedback aware query answering in dataspaces.

ACKNOWLEDGMENTS

The work presented in this paper is funded by Science Foundation Ireland under Grant No. SFI/08/CE/I1380 (Lion- 2).

REFERENCES

[1] T. Heath and C. Bizer, "Linked Data: Evolving the Web into a Global Data Space," *Synthesis Lectures on the Semantic Web: Theory and Technology*, Feb. 2011.

[2] S. O'Riain, E. Curry, and A. Harth, "XBRL and open data for global financial ecosystems: A linked data approach," *International Journal of Accounting Information Systems*, Mar. 2012.

[3] A. Freitas, E. Curry, J. G. Oliveira, and S. O'Riain, "Querying Heterogeneous Datasets on the Linked Data Web: Challenges, Approaches, and Trends," *IEEE Internet Computing*, vol. 16, no. 1, pp. 24-33, Jan. 2012.

[4] S. J. Russell and P. Norvig, *Artificial Intelligence - A Modern Approach: The Intelligent Agent Book*. Prentice Hall, 1995, pp. 1-932.

[5] P. Donmez and J. G. Carbonell, "Proactive learning," in *Proceeding of the 17th ACM conference on Information and knowledge mining - CIKM '08*, 2008, p. 619.

[6] M. Yakout, A. K. Elmagarmid, J. Neville, M. Ouzzani, and I. F. Ilyas, "Guided Data Repair," *Proceedings of the VLDB Endowment*, vol. 4, no. 5, pp. 279-289, 2011.

[7] S. R. Jeffery, M. J. Franklin, and A. Y. Halevy, "Pay-as-you-go user feedback for dataspace systems," in *Proceedings of the 2008 ACM SIGMOD international conference on Management of data - SIGMOD '08*, 2008, pp. 847-860.

[8] W. Fan, "Dependencies revisited for improving data quality," *Proceedings of the twenty-seventh ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems - PODS '08*, p. 159, 2008.

[9] W. Fan, H. Gao, X. Jia, J. Li, and S. Ma, "Dynamic constraints for record matching," *The VLDB Journal*, vol. 20, no. 4, pp. 495-520, Nov. 2010.

[10] A. Elmagarmid, P. Ipeirotis, and V. Verykios, "Duplicate Record Detection: A Survey," *IEEE Transactions on Knowledge and Data Engineering*, vol. 19, no. 1, pp. 1-16, Jan. 2007.

[11] A. Gal and P. Shvaiko, "Advances in Ontology Matching," in *Advances in Web Semantics I*, vol. 4891, no. i, Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 176-198.

[12] J. Bleiholder and F. Naumann, "Data fusion," *ACM Computing Surveys*, vol. 41, no. 1, pp. 1-41, Dec. 2008.

[13] C. Bizer and R. Cyganiak, "Quality-driven information filtering using the WIQA policy framework," *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 7, no. 1, pp. 1-10, Jan. 2009.

[14] O. Hartig, "Querying Trust in RDF Data with tSPARQL," in *Proceedings of the 6th European Semantic Web Conference on The Semantic Web: Research and Applications*, 2009, pp. 5-20.

[15] C. Fürber and M. Hepp, "Using SPARQL and SPIN for Data Quality Management on the Semantic Web," in *Business Information Systems*, vol. 47, no. 1, Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 1-12.

[16] A. Halevy, M. Franklin, and D. Maier, "Principles of dataspace systems," in *Proceedings of the twenty-fifth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems - PODS '06*, 2006, pp. 1-9.