Metadata Reconciliation for Improved Data Binding and Integration

Hiba Khalid[1,2(✉)], Esteban Zimanyi[1], and Robert Wrembel[2]

[1] University Libre de Bruxelles, Brussels, Belgium
{hiba.khalid,esteban.zimanyi}@ulb.ac.be
[2] Poznan University of Technology, Poznan, Poland
robert.wrembel@cs.put.poznan.pl

Abstract. Data Integration has been a consistent concern in the Linked Open Data (LOD) research. The data integration problem (DIP) depends upon many factors. Primarily the nature and type of datasets guide the integration process. Every day, the demand for open and improved data visualization is increasing. Organizations, researchers and data scientists all require more improved techniques for data integration that can be used for analytics and predictions. The scientific community has been able to construct meaningful solutions by using the power of metadata. The metadata is powerful if it is properly guided. There are several existing methodologies that improve system semantics using metadata. However, the data integration between heterogeneous resources for example structured and unstructured data is still a far fetched reality. Metadata can not only improve but effectively increase semantic search performance if properly reconciled with the available information or standard data. In this paper, we present a metadata reconciliation strategy for improving data integration and data classification between data sources that correspond to a certain standard of similarity. The data similarity can be deployed as a power tool for linked data operations. The data publishing and connection over the LOD can effectively be improved using reconciliation strategies. In this paper, we also briefly define the procedure of reconciliation that can semi-automate the interlinking and validation process for publishing linked data as an integrated resource.

Keywords: Metadata · Data reconciliation · Metadata reconciliation Open refine · Data integration · Fuzzy matching · Semantic metadata

1 Introduction

Data reconciliation is generally defined as the validation process of existing data against standardized web services, databases and portals in order to confirm and maintain consistence [2]. The concept of data reconciliation can be easily understood with the example of matching a local movie database with a standardized movie database. In this example, the local database titles and text values can be

S. Kozielski et al. (Eds.): BDAS 2018, CCIS 928, pp. 271–282, 2018.
https://doi.org/10.1007/978-3-319-99987-6_21

validated against an already existing and established database. In order to validate, the reconciliation process performs clustering and fuzzy matching to obtain a similarity index i.e. the amount or level of acceptable matching between the two un-coordinated resources. Reconciliation extends its functionality for linked open data as well. The linked open data is a mechanism for publishing and utilizing structured data over interlinked resources to perform semantic analysis. Linked data integration i.e. the fusion of two independent resources together is based on data complexities. The complexity in linked data world is regarded as the Data Integration problem. For the simplicity of the system there has been set a custom metric called 'Similarity Threshold' that has been set to find data sets that match more closely. The data integration problem comprises many subsets however for the research under discussion it has been restricted to integrating datasets with a similarity index of 75% or above. This similarity index has been inspired based on Jaccard practice. the practice includes the description of objects or group of objects. the object is defined as any constituent in the element set and the overlap is defined as the number of objects that are similar between the sets i.e. union of both available sets. Furthermore the experiment bed evaluates different similarity indices to observe changes in the final results. For instance, one test scenario observes SI of two sets on a threshold value of 45%. This indicates that lowering the SI bar can cause more inclusion of values but can also be under-represented information. This in fact is not the case for all datasets. The SI index can behave differently based on class type. The class type indicates the genre of the dataset under consideration along with the region based classification i.e. how one part of dataset might be representing information with a third dataset. This, brings in the possibility of including more datasets for integration analysis. The similarity index directly coincides with the aforementioned data reconciliation concept. In order to integrate linked data sets more effectively; data and metadata reconciliation play an important role. For the purposes of linked data, the reconciliation process can be easily executed for RDF resources. The RDF Refine primarily focuses on reconciliation based on SPARQL endpoints and RDF dumps. It particularly looks and scans the web for RDF datasets and resources to perform a validation process. Metadata reconciliation operates in a similar manner. The designed reconciliation API actually connects the collection or dataset specific vocabulary with a standardized vocabulary provided over the internet. The metadata connections can be monitored and controlled to perform statistics based validation. This requires the API to perform a similarity index matching either by using a value to key based clustering technique or a fuzzy matching technique. In this way, the metadata reconciliation provides meaning to the values in field such as text. The meaning also enhances the metadata interpretation by the foreign or unrecognized datasets that might contain similar information. A simplified overview of reconciliation process is described in Fig. 1. The approximate string matching technique i.e. fuzzy matching operates by finding pattern approximation. The approximate string matching technique is composed of two main subsets:

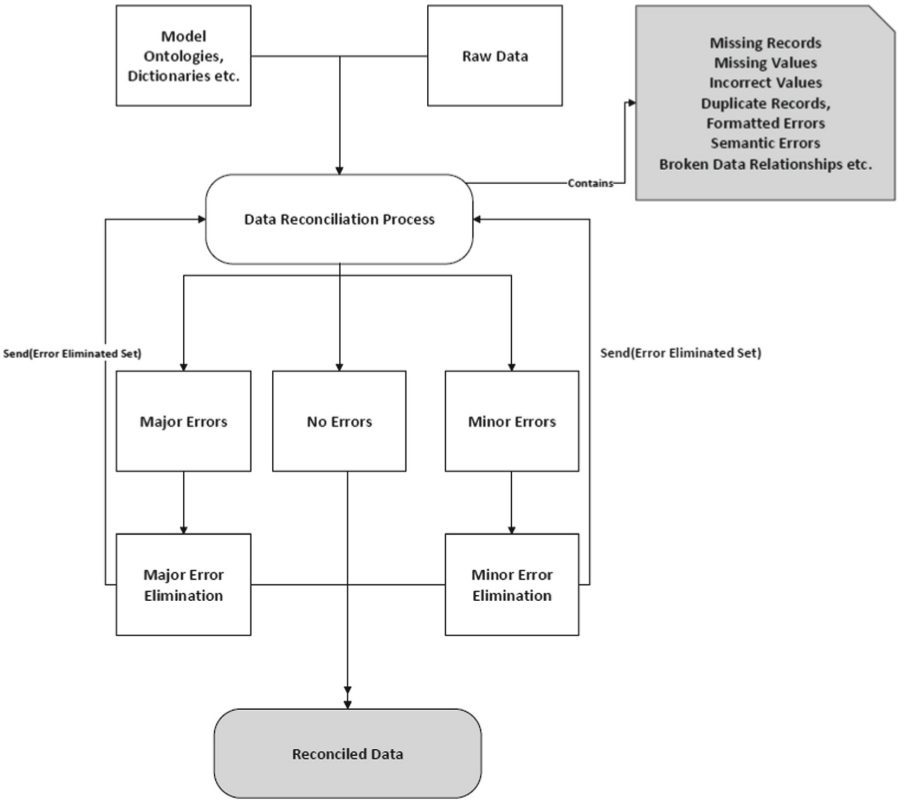

Fig. 1. Reconciliation process overview

1. Finding an approximate match for a given test set.
2. Finding a dictionary based pattern approximation.

The rest of the paper is organized as follows: The Sect. 2 discusses and elaborates on research techniques studied and existing scenarios to understand the problem domain. Section 3 addresses the proposed metadata reconciliation for data integration problem. Section 3.1 addresses the elaborate discussion on metadata representations. The Sect. 4 summarizes the paper and discusses the future work.

2 Related Work

Publishing linked data is one of the most important factors in the problem of data integration. Many scientists and researchers are evolving techniques and tools to facilitate the process of automatic linked data publishing. Data reconciliation is one of the techniques explored by data scientist in automating or more precisely semi-automating the process of linked data publishing. The first

step in this process is to identify resources or datasets that are similar in context and can be integrated. The research community experiments with live filtering of DBpedia content such as utilizing live filtering [2] and extraction of data as well as metadata for creating validated datasets. These datasets are then published on the DBpedia. This is effectively accomplished using live DBpedia by incorporating rules [2] for filtering and update processes. There are other techniques that focus primarily on organizing and analyzing big ontology data. One such problem has been addressed with the use of Hadoop [3]. The classical model representation of web service management using Hadoop [8] can be utilized. The idea and methodology of parent to child node can be used for events and entities and even attributes in relevance to the perturbed study [8]. These relationships can then be in return used to explain the distance between tow nodes or distance between tow parent classes i.e. identifying similarities between sub-data categories. The web-services can be assumed as class based sets that can have parallel instances and operative measures. This is accomplished by establishing distributed ontology clusters and then setting up a basis for more accurate linking and reconciliation strategies. The main concept behind devised and evolving techniques is the identification of datasets that correspond to each other. This is called relative measure. The datasets can only be properly published and integrated if their similarities and differences are adequately quantified. It effectively uses the framework developed for visualization of DBpedia ontology and metadata visualization. Many recommendation [5] algorithm based approaches have been introduced by data scientists for processing non-trivial information over the linked data. Freebase knowledge available in web of data specially in DBpedia and Wikipedia data has been annotated for entries [5] to produce better ranking and recommendations in the web of data documents. Different approaches have been addressed towards reconciliation and relativity identification. The LDA algorithm [1] has been used for approximate string matching previously indicating promising results. This similarity can be easily deployed on linked data sources to reflect similar concepts. The LDA algorithm contributes towards various forms of inputs. The technique is not specific to text or value based sets. The technique is very effective in graphic [4] and image based sets. The multipurpose use of LDA algorithm allows it be experimented with different techniques. For the usefulness of algorithm with its simplicity is the main reason of effective and accurate results. In a similar context Wikipedia [2] is one of the biggest examples. The use of events and elaboration from these events can provide insights such as observation of connected entities based on order of appearance or based on order of connection establishment. The standard deviations along with normal distributions for events can identify the lag pertaining to category or class. in a similar methodology context the DBpedia [6] is responsible for extracting information from Wikipedia and making it publicly available for third party use. There are many factors that contribute in integration and matching. One such very important factor is the need of revision based changes and updates [6] of integration based on reviewed or updated changes. The mappings need to be up to date and relevant for entities to correspond and develop new class

relations [2,6]. Based on Wikipedia data automatic ontology can also be built if relations are clearly defined [7]. Suing search queries is an interesting concept. The search queries are text-based thus a relative query equivalence can be derived from them providing insight into how word matching and similarities can effect ontology based matching results [7].

3 Methodology

Problem Statement: The broadest sense of problem is the domain problem is managing data integration using metadata. The hypothesis of the research study is subjected on using metadata as a driving catalyst in the process of data integration. This data integration can be deployed on both traditional relational data sources and linked data sources. In the context of this research contribution, the proposed methodology is experimented on the hypothesis of using reconciliation methodology for improved data quality along with the use of approximate string matching to attain a more information, complete and reflective set of metadata. The methodologies imposed for experimentation include the use of LDA, and N-Gram algorithms for locating approximate distances and using fuzzy string matching with Jacard similarity index for calculation the relation between two sets. This is regarded as a coherence or the percentage of similarity that might exist between two nodes or two columns in a dataset. (The number of columns is not restricted to two. Jacard index and similarity index for column text matching is run from column 1 to column n, where n is the maximum number of columns.) Data reconciliation presents an opportunity for managing linked open data using metadata. Metadata alone without enrichment and classification can only serve restricted purposes and operations. However, if the concept of reconciliation can be incorporated the process outputs and functions can increase. The study under discussion involves the demonstration and experimentation of Open Refine technologies to improve data reconciliation before publishing data over the linked data network or repository. The study conducted has utilized two standard datasets for experimental purposes. The first dataset is based on amazon reviews and the second dataset is just a collection of NIPS research papers from 1987 to 2015. Both of the test datasets have been retrieved from the UCI repository (https://archive.ics.uci.edu/ml/index.php). The research focuses on enriching and finding relations between metadata sets. This encourages the purposeful deployment of traditional data as lined data. If metadata can be properly enriched and classified than relative relations between linked open data and legacy or traditional data can be easily enhanced. Thus, the primary focus of this research experimentation is to understand the enrichment process of metadata to provide a ground for linking it with linked open data. Open Refine is a n open source initiative to provide simpler and manageable solutions for messy datasets. One of the most widely used property of Open refine is its ability to perform approximate clustering for value standardization. Another important feature for cleaning and standardizing data before publishing on the linked data portal is data Enrichment. This not only helps in closing the gap but also increases the

efficiency in establishing links over the LOD. The general method for reconciliation comprises the following steps:

1. Column identification from home datasets for reconcile process. This process involves the identification of all possible columns that can have a match with a standardized vocabulary. Refer to Fig. 2 to understand how each attribute can be matched to find individual similarity and collective similarity of the daatset. In the Fig. 3 datasets have been considered to illustrate the rule of associative matching i.e. if two distinct datasets have one dataset in common an inference can be derived. Set A, B and C are three datasets and B is paired for matching with A and C i.e. (A, C) and (B, C) then if their similarity index is above the threshold value say 75%. An associative relation or inference can be derived that set A and C might have a relation as well i.e. (A, C) similarity can be infered.
2. Define a collection sub-division for identifying the match space.
3. Identification and selection of a standardizes vocabulary. There are several vocabularies that have been designated to provide metadata matching and reconciliation properties. An important step is to identify the domain of metadata under consideration. If the domain of metadata is well known and established a vocabulary can be easily downloaded for clustering, reconciliation and standardization. Vocabularies with already existing SPARQL endpoints are easier and more manageable to operate upon as compared to manual vocabularies.
4. Data vocabulary broadcasting and loading. This step particularly involves the addition of vocabulary to the open refine environment. The process of uploading a vocabulary is attained by using the reconciliation service. For vocabulary upload the RDF extension for open refine has to be installed. This is mandatory for the system to openly communicate with the vocabulary and perform necessary operations such as matching and search space legitimization. After installation of RDF extension, the library can be added using the reconciliation service and by live SPARQL endpoint monitor.

1. Finding an approximate match for a given test set.
2. Finding a dictionary based pattern approximation.

The mapping values with corresponding columns are then reconciled for purposeful use and operation. The matching criteria is referred to as a facet matching phase. This particularly focuses on identifying the matches acquired through the reconciliation process. The extent of match is predefined into two categories namely: matched or unmatched. A third research prospect introduced is the similarity index. This indicates whether a match was in the bounds of 75% to 95%. This is regarded as an approximate match. For our project research the approximate match has been acquired through clustering technique and fuzzy matching. The clustering technique accommodates the errors and matching discrepancies in the syntax design or writing prospects. In relevance to semantic discrepancies a freebase semantic library has been effectively enriched to accommodate discrepancies more than just syntax. A collection of algorithms has been defined in

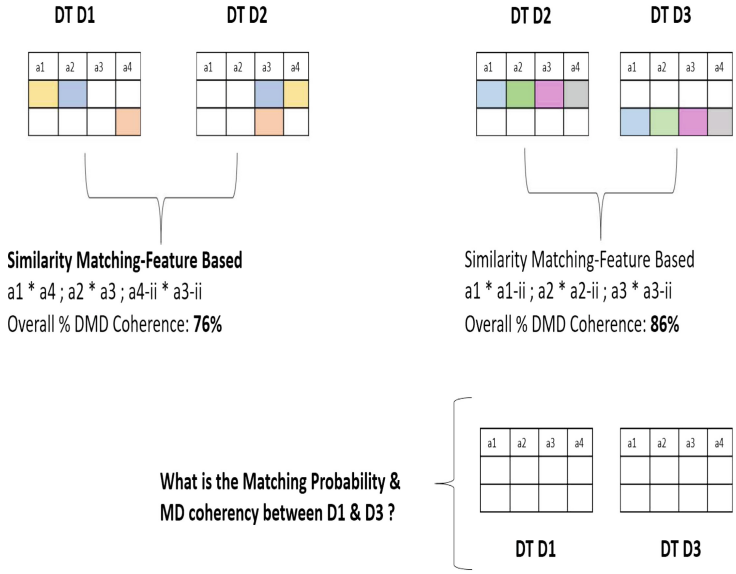

Fig. 2. Illustration of column or attribute matching scenario

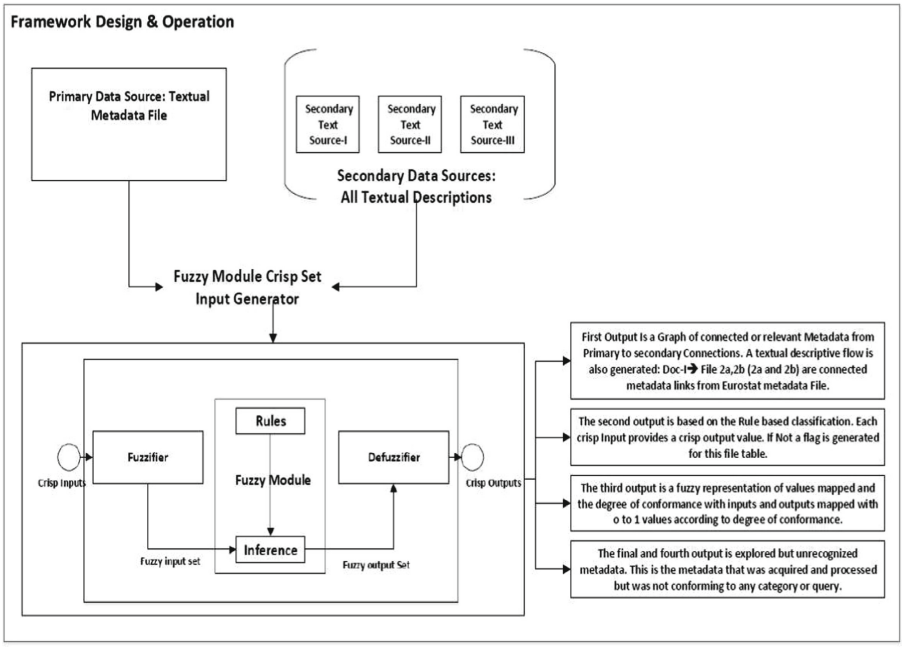

Fig. 3. Simplified overview of fuzzy role in reconciliation and approximation module

addition to key collision methods. A key collision method has been incorporated to introduce the concept of multiple data value representations. Each data value i.e. string or text in the conducted study has some key values that directly relate or are meaningful. A set of collision buckets are created to collect relevant yet different keys. This can create key collision in the system. The system has been designed on linear complexity methodology thus experimentally 2 million rows of data can be effectively classified without a single occurrence of collision. This is attained by using a cluster for all key values and all supported or possible collisions. For the research under discussion, the approximate string matching is acquired using three popular methods.

3.1 Levenshtein Distance Algorithm (LDA)

The primary purpose of LDA is to find or locate a difference connection between two sets of sequences. The task is acquired through basic database operations like insertion, deletion or updating. Each operation like in traditional databases is achieved through the set of command sequences. The minimum distance is computed with certain criteria based methods such as described below:

1. Initial state: this here indicates the keys or words or rows that require a match. For example, for a NIPS research paper the name roher and Roher can be put through a similarity find to see if the two authors are same or not.
2. Operators: operators on computation perform the task of insertion, deletion or substitution often regarded as update procedure in row based operations for relational databases.
3. Goal State: this identifies the final word for match that the library is trying to identify for validation.
4. Path cost: this is identified as the cost minimization rule for the collision. This is attained by minimizing the number of edits per search.

Leveinshtein Distance Algorithm

```
Public Class Distance(private int Minimum (int a, int b, int c)){
int mi; mi = a;
if (b < mi) {mi = b}
if (c < mi) {mi = c}
return mi;
}
Public String LD (String s, String t){
int d [][]; //Matrix
int n; //Length of S
int m; // Length of T
int i; // Iterates Through S
int j; //Iterates Through T
Char s_i; // ith Character of S
Char t_j; //jth Character of T
int Cost; //Cost
```

```
n = s_length();
m = t_length();
if (n == 0) {return m;}
if (m == 0) {return n;}
d = new int[n+1][m+1];
    for (i=0; i<=n; i++) {d[i][0] = i;}
    for (j=0; j<=m; j++) {d[j][0]= j;}
    for (i=1; i<=n; i++) {s_i = s.charAT (i-1);}
    for (j=1; j<=n; j++) {t_j = t.charAT (j-1);}
if (s_i == t_j) {cost = 0;}
else {cost = 1;}
d[i][j] = Minimum {(}+1 , d[i-1][j-1] + cost);}}
return d[n][m];
}}
```

3.2 Damerau Levenshtein Distance Algorithm (DLDA)

The DLDA is similar to LDA with an additional feature for transposition. The transposition feature allows for a character in the matching string to be converted into another character thus increasing the match criteria's and benefiting for approximate string matching. The cost for this function is computed by minimizing the number of insert, delete, update and transposition for converting one word to another for string matching. This algorithm is most useful to identify fraud records or fraud URIs.

3.3 N-Gram

N-gram is identified as a subset contiguous sequence from a provided sequences of words, text or speech. For the limitations of this project the primary mode of development has only been restricted to text based inputs for incorporation and reconciliation with publishing linked data. The n-gram can accept a wide variety of inputs such as base letters, characters, words, syllables and phenomes. We have only considered base pairs, letters and words for identification from the two datasets and for vocabulary matching purposes. N-gram has been introduced with open refine to incorporate the ability of predicting next text in the continuous sequence. It is attained by using a Markov model for a sequence of text type (n−1). The limitation of research currently is the unrecognized text formats. This will be effectively handled in the next modules of the working project. Reconciliation is an important step towards publishing data over the linked data. The tool that has used for importing and exporting with Open Refine is Linked Media Framework. The tools provide an easy alternative for connecting legacy data in the formats of traditional excel, CSV and tables format onto the linked data network. Figure 4 presents an overview of the system generated to connect with DBpedia and its vocabularies with relevance to datasets under consideration.

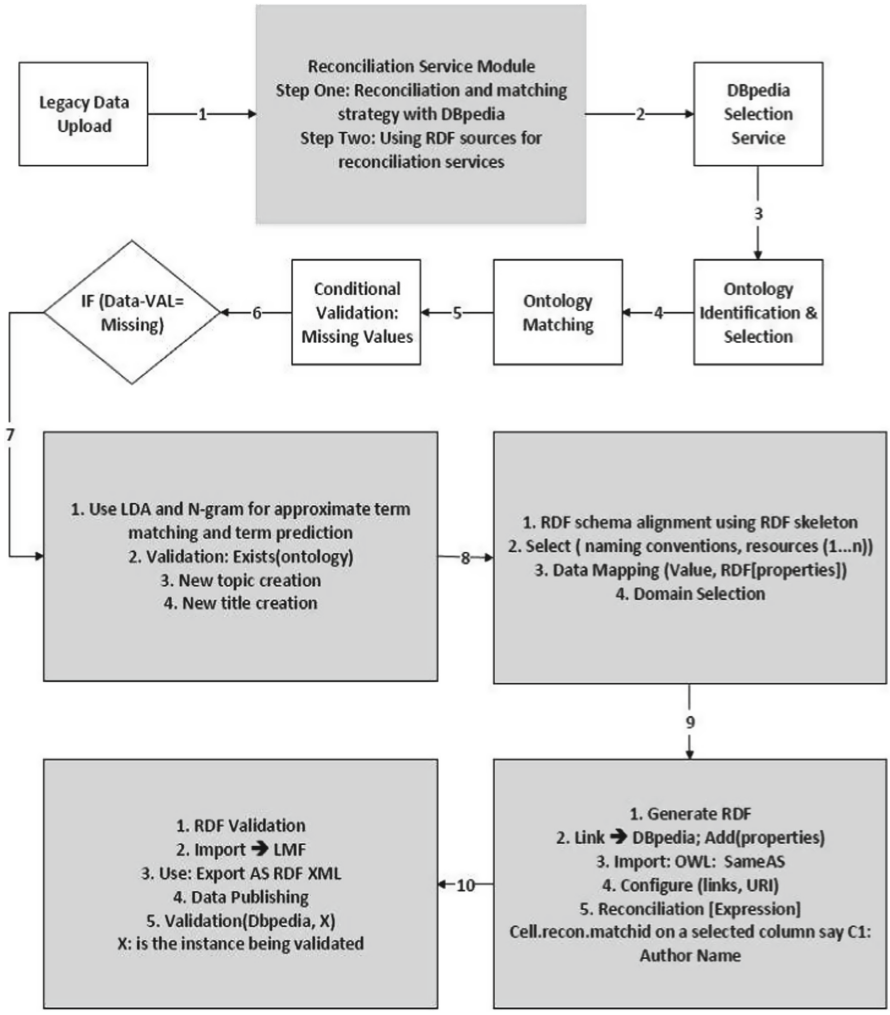

Fig. 4. Publishing legacy data as linked data using reconciliation properties

3.4 Results and Analysis

The hypothesis was tested based on different experimental settings. The most optimal and relevant to this research contribution is based on similarity index of 75% or above. For each data class iteration the results were observed in case of each algorithm. The Fuzzy algorithm clearly has advantage over the other algorithms, the controller used for Fuzzy is called the Mamdani fuzzy controller. The values and approximations are classified from ranges between 0 and 1. Thus, the approximation and similarity approximation for fuzzy is higher than the rest of the algorithms. In combination based matching fuzzy performs optimal with LDA. A summary fo results have been provided in the table below. To understand

the results the value of input must be clearly understood. For each evaluation iteration 70000 values were considered and there were a total of 50 sample class cycles. The measures provided below are generated based on average of all class cycles. The algorithms LDA, DLDA, N-gram were computed in python. The fuzzy Mamdani controller was also computed in python and evaluated over Weka (Table 1).

Table 1. Evaluation results for LDA, DLDA, N-Gram, fuzzy

Technique	Recall	Precision	Accuracy
LDA	85	84	87
DLDA	87	81	89
N-gram	78	74	80
Fuzzy	88	85	91

4 Conclusion and Future Work

The scientific community is working everyday to develop tools that can bridge the gap between automatic data exploration, curation and upload processes. Based on the detailed survey of tools conducted for experimenting and locating tools that can perform automatic or even semi-automatic publishing standards, no such tool has been produced at the moment. The most respected communities in linked data and metadata standards such as statistical metadata exchange, linked data and Dublin core all provide solutions such as methods to write a publishable data and standards, policies for developing metadata. To the best of our scientific knowledge and survey there is no automatic method for publishing legacy data as linked data. However, many semi-automatic methodologies and tool exist to minimize the manual working such as Open Refine developed by Google can be used to perform many tedious tasks on datasets of samll sizes such as reconciliation. The reconciliation is however limited to exporting RDF sources. The extension has been developed to support the needs in RDF community for reconciliation and managing sources. One such tool with additional libraries and self-created methodology has been utilized for the research purposes. Open Refine and Linked Media Framework (LMF) provide the facility of publishing legacy data as linked data using reconciliation as a primary factor of linking. For the conducted experiment the limitation was placed upon the type of data. Only textual data has been considered at the moment for semi-automatic data publishing. The scientific findings also include the conclusion that not all data and metadata can be in one format. Thus, developing a complete general purpose software for automatic data upload is both unrealistic and very costly. Dublin Core, LMF are research based and experimental developments that are currently working in research for past 6 years to develop methodologies for targeted concerns such as labelling of metadata, policies on generalized formats,

the methods of writing and generating metadata etc. The analysis presents an opportunity for incorporating increase in an overlaying metadata reconciliation layer to enhance the property functions by a value of 30%. The research has experimented and discussed the methods that are deployed for managing metadata connection and for validating the input as well. The future of research is to develop a web service for extraction and enrichment of published metadata to generate accurate connections or linking properties. This also includes the expansion of domain set from text to multivariate values. Another prospect of future research development is to introduce a metadata framework that can reconcile and also publish data by using fuzzy matching for finding triples or pairs of similar metadata sub-sets.

Acknowledgments. This research has been funded by the European Commission through the Erasmus Mundus Joint Doctorate Information Technologies for Business Intelligence-Doctoral College (IT4BI-DC).

References

1. Amir, A., Lewenstein, M., Porat, E.: Faster algorithms for string matching with k mismatches. J. Alg. **50**(2), 257–275 (2004)
2. Fetahu, B., Anand, A., Anand, A.: How much is Wikipedia lagging behind news? In: Proceedings of the ACM Web Science Conference, p. 28. ACM (2015)
3. Georgescu, M., Kanhabua, N., Krause, D., Nejdl, W., Siersdorfer, S.: Extracting event-related information from article updates in Wikipedia. In: Serdyukov, P., et al. (eds.) ECIR 2013. LNCS, vol. 7814, pp. 254–266. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-36973-5_22
4. Ho, T., Oh, S.R., Kim, H.: A parallel approximate string matching under Levenshtein distance on graphics processing units using warp-shuffle operations. PloS One **12**(10), e0186251 (2017)
5. Lehmann, J., et al.: DBpedia-a large-scale, multilingual knowledge base extracted from Wikipedia. Seman. Web **6**(2), 167–195 (2015)
6. Morsey, M., Lehmann, J., Auer, S., Stadler, C., Hellmann, S.: DBpedia and the live extraction of structured data from Wikipedia. Program **46**(2), 157–181 (2012)
7. Ochs, C., Tian, T., Geller, J., Chun, S.A.: Google knows who is famous today-building an ontology from search engine knowledge and DBpedia. In: 2011 Fifth IEEE International Conference on Semantic Computing (ICSC), pp. 320–327. IEEE (2011)
8. Zhu, X., Wang, B.: Web service management based on Hadoop. In: 2011 8th International Conference on Service Systems and Service Management (ICSSSM), pp. 1–6. IEEE (2011)