Consistent Process Mining Over Big Data Triple Stores

Antonia Azzini, Paolo Ceravolo
SESAR Lab - Dipartimento di Informatica
Università degli Studi di Milano, Italy
Email: {name.surname}@unimi.it

***Abstract*—'Big Data' techniques are often adopted in cross-organization scenarios for integrating multiple data sources to extract statistics or other latent information. Even if these techniques do not require the support of a schema for processing data, a common conceptual model is typically defined to address name resolution. This implies that each local source is tasked of applying a semantic lifting procedure for expressing the local data in term of the common model. Semantic heterogeneity is then potentially introduced in data. In this paper we illustrate a methodology designed to the implementation of consistent process mining algorithms in a 'Big Data' context. In particular, we exploit two different procedures. The first one is aimed at computing the mismatch among the data sources to be integrated. The second uses mismatch values to extend data to be processed with a traditional map reduce algorithm.**

***Keywords*-Data Integration, Process Mining, Big Data**

I. Introduction

The diffusion of the 'Big Data' paradigm was mainly motivated by the need of processing large data sets running various resources in parallel. However, the focus on distributed systems and lack of schema support pushed its adoption also in cross-organization scenarios, where the aim is integrating multiple data sources to extract statistics or other latent information. In general data integration can be implemented both by imposing a common conceptual model, which provides a uniform space to resolve resource names, or by a peer-to-peer approach where no common model is imposed to the data sources [1]. Although, in a 'Big Data' context, when volume of data exceeds the organization storage or compute capacity for accurate task execution, agreeing on a common model is often an necessity, as the execution time cannot cope with the effort of resolving the variety of semantic representations. When data variety involve heterogeneous and unstable representations the Resource Description Framework (RDF) data format has gained popularity mainly because of its ease of use and flexibility. Lineage information, as well as metadata or annotations are expressed in an uniform format, additions are monotonic, context can be represented, resolvable URIs can be used to retrieve information via HTTP lookup [2]. However, the adoption of a common and monotonic data model gives answers to issues related to data interoperability and data evolution while the implications of semantic heterogeneity are actually wider [3]. This can be easily understood considering that each local organization independently apply a semantic lifting procedure for expressing the local representation in term of the common model. Clearly, this procedure may differ in each local source, as discussed in Section IV. The result is the introduction of potential inconsistencies when different resources are erroneously processed under the same name.

In this paper we consider the implications of this issue for process mining techniques that are based on the computation of frequency among event dependency. We illustrate a methodology exploiting two separate procedures. The first procedure is aimed at computing the mismatch among the data sources to be integrated. The second procedure uses this information for executing a map reduce algorithm prepared to integrate data in a consistent way. In particular in Section II we introduce the motivation of our work and the scenario used to guide the discussion, in Section III we describe the technological basis we have considered, in Section IV we discuss the semantic lifting problem that is the central problem we deal with, and we collocate this problem in the 'Big Data' context, in Section V we illustrated our proposal and in Section VI we go to the conclusions.

II. Scenario

In this work we focus our attention on Process Mining techniques. To illustrate our proposal we present a scenario related to event tracking in social communities. As known, everyday social media provide a formidable trail of human activities. The information collected from social media is exploited to profile users according to their preferences or behavior, providing a capital benefit to a wide range of applications, such as advertising, social recommender systems, and knowledge management. However, advanced applications in the area requires to integrate the information stream generated from multiple sources.

In order to predict user behavior it is possible to exploit a wide range of techniques, including machine learning, text mining, human-computer interaction, and social science. Current state of the art include techniques ranging from data analytics [4] to complex event processing [5]. One specificity of process mining techniques is that they are mainly base on constructing dependency/frequencies tables. This focus on frequencies allow to elaborate data exploiting the composition of aggregate quantities. However, these

techniques cannot cope with the characteristics of 'Big Data' [6], as they traditionally assume stable process structures and a limited, a priori fixed number of processes. For instance, according to [7] process management in the large demands the selection of relevant events and tracks, the correlation of relevant events to connect the traces of hybrid processes, and clustering techniques to detect process changes and distinguish process structures. In this work we underline that a typical problem of the integration of distributed data sources is handling the semantic lifting procedures applied locally, as discussed in detail in Section IV.

In order to illustrate the impact that an appropriate semantic lifting can have on process mining we propose an example based on the characterization of a user behavior scenario. In this case we envisage a system collecting social media contributions for characterizing user behavior. In order to cope with this kind of analysis it is important to identify the expected behavior for specific classes of interactions that can be defined according to relationship, content type, topics or others categories. Such an information can be of paramount importance for improving the design of social media or to make more effective the recommendation or guidance of the content shared by social media. The common model of our scenario can be realized by a standard RDF vocabulary for modeling activities in online communities such as the SIOC model [8] that is illustrated in Figure 2.

III. RDF Graphs in the Big Data context

Generally speaking, the RDF corresponds to a standard vocabulary, defined at the basis of the Semantic Web [9], and composed by three main elements: concepts, attributes and relations between them. These elements are modelled as a labelled oriented graph [10], defined by a set of triples $< s, p, o >$ where s corresponds to the subject, p to the predicate and o is the object, combined as shown in Figure 1.

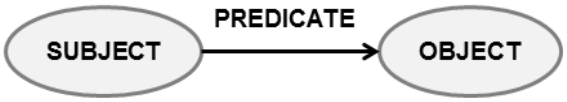

Figure 1. Example of RDF subject-object relation.

An important characteristic of this model is that additions are monotonic. New information is added into an RDF graph by appending new triples to the list. Such representation allows big benefits for real time business process analysis: data can be appended 'on the fly' to the existing one and it will become part of the graph, available for any analytical application, without need for reconfiguration or any other data preparation steps. Moreover, an RDF provides a basic set of semantics, used to define concepts, sub-concepts, relations, attributes, but also context and annotations and it is easily extendible with any domain-specific information.

RDF is an extremely generic data representation model that can be used in any domain. Its standard vocabularies allow external application to query data through SPARQL query language [11], a standard language based on conjunctive queries on triple patterns as graph views, that are able to identify paths in the considered graph. SPARQL is also supported by most of the triples stores available [12]. Moreover, as reported in [13], RDF tripe stores are more and more faced with very large graphs that contain hundreds of millions of RDF triples. The ZEUS framework represents one of the most advanced solutions specifically designed for integrating multiple source and supporting fast and continuous execution of SPARQL queries [14].

As said, RDF brings several advantages for data integration. Nevertheless, it uses a verbose data format and it implies a hierarchical structure to the data so that the queries connecting different entities may potentially involve long chain of joins. In this case, when such algorithms need to be applied over 'Big Data', dimension reducing algorithms allow to reduce the datasets dimensions for each company. Several solutions have been proposed in the literature [15], [16], [17], and, among them, the so-called 'map-reduce' algorithm represents a suitable, and well known approach, for data integration [18], [19].

IV. Semantic Lifting

By semantic lifting we refer to all the transformations of low-level systems logs carried out in order to achieve a conceptual description of business process instances. Typically this procedure is implicitly done by converting data from the data storages of an information system to an event log format suitable for process monitoring [20], even if, in the literature, several approaches based on triple stores were also proposed [14] [21]. Table I shows a fragment of a workflow log of the posting activities tracked in a social community site. The system reports all the events related to a specific post, spotlighting the content type used within each post. The first choice to be done for interpreting data regards the identification of the workflow instances. In our example the sequences of events are grouped according to posts created in the community. Now, data can be analyzed by using process mining algorithms, that are based on detecting ordering relations among events, in order to characterize a workflow execution log [22]. This way a single execution can be compared to verify the satisfiability of specific conditions according to the order of the event executions. In particular the key notion corresponds to the notion of successor.

Given a set of traces or instances T of a workflow W, if two events a, $b \in W$, we have $a \succ_W b$ if and only if there is a trace $t : \{e_1, e_2, e_n, \}$ and $e_i \equiv a$ then $e_{i+1} \equiv b$. Similarly we use the notation $a \succ^n b$ to express that an event b is successor of an event a by n steps. By using these notions we can construct dependency/frequency tables that

Table I
LOG DATA OF THE SOCIAL COMMUNITY.

Event	User	Timestamp	Content Type
Post AAA			Text
Create	userP	2012-11-09 T 11:20	
Reply	userV	2012-11-09 T 19:20	
View	userP	2012-11-09 T 19:22	
Post AAB			Text
Create	userP	2012-11-09 T 11:20	
Reply	userA	2012-11-12 T 10:23	
Delete	userP	2012-11-14 T 18:47	
Post AAC			r1.com
Create	userP	2012-11-15 T 12:07	
Reply	userM	2012-11-18 T 09:21	
View	userP	2012-11-18 T 14:31	
Post AAD			r2.com
Create	userF	2012-12-03 T 09:22	
Reply	userG	2012-12-03 T 12:02	
Reply	userL	2012-12-03 T 17:34	
View	userV	2012-12-05 T 11:41	
Post AAE			r3.com
Create	userD	2012-12-05 T 11:41	
Reply	userD	2012-12-08 T 10:36	
Reply	userV	2012-12-08 T 16:29	
View	userD	2012-12-05 T 16:58	
Post AAF			r4.com
Create	userG	2012-12-10 T 08:09	
Reply	userV	2012-12-10 T 18:38	
Reply	userF	2012-12-10 T 18:38	
Delete	userG	2012-12-10 T 18:38	
Post AAG			Text
Create	userV	2012-12-04 T 10:26	
Post AAH			Text
Create	userV	2012-12-04 T 13:12	
Reply	userG	2012-12-04 T 15:22	
Reply	userD	2012-12-04 T 16:21	
View	userV	2012-12-04 T 16:45	
Post AAI			Text
Create	userV	2012-12-05 T 10:12	
Post AAL			r5.com
Create	userA	2012-12-05 T 12:22	
Reply	userD	2012-12-06 T 14:51	
Reply	userM	2012-12-07 T 10:31	
View	userA	2012-12-05 T 13:08	
Post AAM			Text
Create	userV	2012-12-04 T 10:26	

allow to verify the relations that constraint a set of log traces. Table II shows the frequencies for all the combinations of events recorded in the data log illustrated in Table I. Such frequencies allow us to verify that the following constraints hold in W: $Create \succ_W Reply$ or $Create \succ^*_W View \succ^*_W Delete$. The successor relationship is rich enough to reveal many workflow properties, but to better characterize the significance of dependency between events other measures based on information theory are adopted in the literature, such as for instance the J-Measure proposed by Smyth and Goodman [23] that quantify the information content of a rule.

However, in order to identify significant constraints on the analyzed sequences, the interpretation on data cannot be neutral. For instance, if we are interested in verifying constraints on events subsequent to a post, by distinguishing posts according to the content type, we cannot relay on Table I. Indeed, the table is sparse and, consequently, few constraints can be proved to hold in W. Constraints on posts with text as content do not demonstrate high frequencies, while constraints on post with other contents do not have a strong support. This sparsity is typical of so called 'spaghetti-like processes', i.e. unstructured processes where recurrent event's sequences are not so easily defined [24].

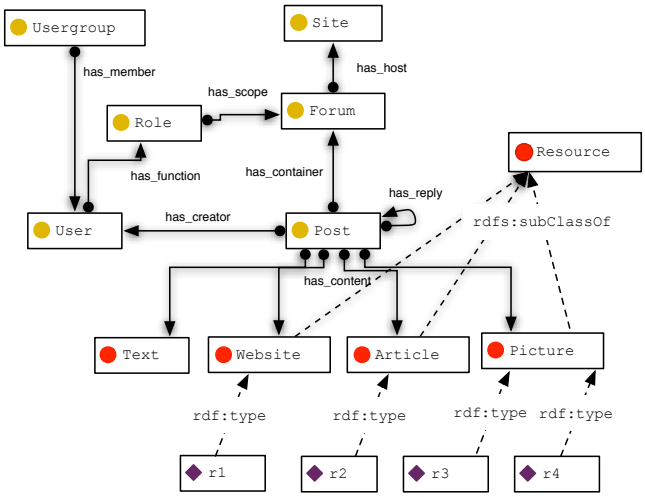

Figure 2. Online Community Model.

A semantic lifting procedure can be applied to the log data for remodeling the representation of the process and implementing additional investigations. In our example we apply the Online Community model described in Figure 2 to our log in order to distinguish among different type of resources posted.

Data interpreted by the semantic lifting allow the definition of Tables II and III. Then, we have a view about the dependence frequencies among events. We can observe that new constraints hold, like: $Create - Resource \succ_W Reply$ in Table II, $Create - Picture \succ^*_W Delete$ in Table III. An informal interpretation of such constraints is that we have an higher probability of a reply to a post containing a resource than one containing only text. Posts with reply are more likely to be deleted by the creator of a picture than other content.

The RDF model is particularly suitable to associate data with complex data models. Moreover, RDF allows to easy aggregate data by considering their shared properties. For example, as shown in Table I, all the log event with the same content type could be grouped together, by defining sets of data with similar properties. For instance, in our example, by considering the `subClassOf` relation we can aggregate data of the same type, by grouping all the content

Table II
EVENT FREQUENCY.

Dependence $a \succ b$	#	Event Frequency					
		Reply		Delete		View	
		$\# \succ$	$\# \succ^2$	$\# \succ^2$	$\# \succ^3$	$\# \succ^2$	$\# \succ^3$
Create-Text	6	3/6	1/6	1/6	0	1/6	1/6
Create-Resource	5	5/5	4/5	0	1/5	1/5	3/5

Table III
EVENT FREQUENCY.

Dependence $a \succ b$	#	Event Frequency					
		Reply		Delete		View	
		$\# \succ$	$\# \succ^2$	$\# \succ^2$	$\# \succ^3$	$\# \succ^2$	$\# \succ^3$
Create-Text	6	3/6	1/6	1/6	0	1/6	1/6
Create-Website	2	2/2	1/2	0	0	1/2	1/2
Create-Picture	2	2/2	2/2	0	1/2	0	1/2
Create-Article	1	1	1	0	0	0	1

values that have a link to a resource, as done in Table III. Moreover, standard techniques for mapping RDF data [25] enable us to link log data to the data model by defining type assignments, as specified in Figure 2. SPARQL queries allow us to manipulate data to view them in the appropriate structural order, by defining, for example, events that are grouped by content.

A. Semantic Lifting in the Big Data Approach

As previously reported, the example considers the integration of data on online communities aggregated from different sources.

Data integration requires different parties to agree on a common model, providing an uniform interface to query the different interconnected data sources. This implies that the mediators from the local systems logs to the common model are independently carried out on the different data sets. Consequently, also the semantic lifting are independently developed over the local systems logs; this aspect can introduce semantic mismatch that is not considered in the data integration process. Figure 3 illustrates this condition. The problem arising from such a scenario is that the quality of the information extracted from integrated sources is compromised. In addition, if the context regards the integration of large data sources, whenever an organization ability to handle, storing and analyzing data exceed its current capacity, the solutions proposed are required to maintain computational complexity under control. For this reason in Section V we propose countermeasures to deal with semantic lifting technologies, that are suitable for business process mining techniques.

V. THE PROPOSED APPROACH

Our aim is to investigate solutions for handling semantic lifting without compromising the requirements imposed in a 'Big Data' context.

The fist step regards the characterization of the semantic mismatch that can derive from distributed semantic lifting procedures. All the data sources share the same common

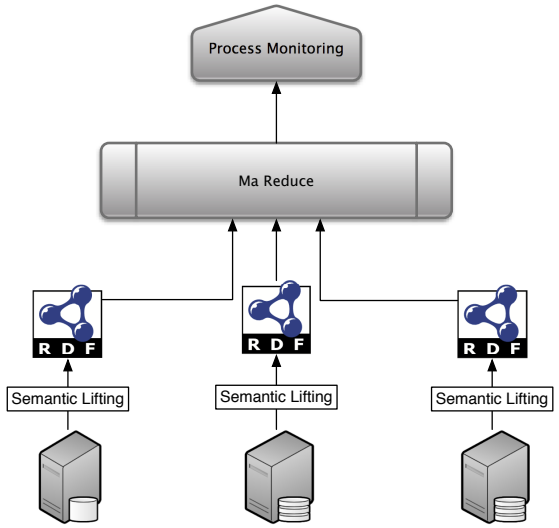

Figure 3. Semantic lifting collocation in distributed data integration scenarios.

model but the interpretation of the predicates adopted in the model can differ.

Given a data set $\mathcal{D}$ composed by a set of triples $< s, p, o >$ we define instance assignment as unary predicates on constants, in the form $P(a)$ or $Q(b)$ and concepts as unary predicates on variables as $P(x)$ or $Q(y)$. To the sake of simplicity in the following we are going to refer to concepts by using capital letters such as: P, Q or R. As we assume that all the data sets share a common vocabulary L, we can express the semantic mismatch among the different interpretations in term of set relations among the concepts of L in the data sets $\bigcup_i^n \mathcal{D}_i$ we are integrating. In particular we are interested in understanding if for $\Re(P_{\mathcal{D}_1}, P_{\mathcal{D}_2})$, $\Re$ is equivalent to 'is included or equal', $\subseteq$, 'includes or equal', $\supseteq$, or 'is disjoined', $\dot{\vee}$. Knowing these relations and taking a dataset $\mathcal{D}_r$ as a reference model we can say that the predicates $\bigcup_i^n P_{\mathcal{D}_i}$ do not create mismatch for

$P_{\mathcal{D}_r} \subseteq P_{\mathcal{D}_i}$, while we have a semantic mismatch in all the other situations. For instance, in case of $P_{\mathcal{D}_r} \dot{\vee} P_{\mathcal{D}_i}$ the integration is not feasible while in case of $P_{\mathcal{D}_i} - P_{\mathcal{D}_r} > 0$ a semantic mismatch is created as some members of $P_{\mathcal{D}_i}$ cannot be integrated with $P_{\mathcal{D}_r}$.

A. Countermeasures to semantic mismatch

In general when a mismatch for a classes is identified, the more conservative way of handling their instances is to classify them under the first superclass where we know semantic mismatch does not apply:

$P_{\mathcal{D}_r}(x) \cup P_{\mathcal{D}_i}(y) \rightarrow y \in Q_{\mathcal{D}_r} if P_{\mathcal{D}_i} \subset Q_{\mathcal{D}_i}$.

This approach could be carried out by SPARQL implementations allowing to relax query results. An example was given in the literature by Reddy and colleague in [26], where the authors proposed a trust-based model for realizing queries by considering the last common ancestor of the predicates involved in the query and in the data store. Each triple asserts the relationship between the subject and the object which is described by the predicate. The trust model assigns scores to each triple pattern that indicates the degree of trustworthiness of the relationship asserted by the triple. A high trust score means that the consumer has a high degree of faith in the information contained in the triple and vice-versa. The trust scores are assigned to the triples by the information consumer based on his subject belief after assessing the triples. The trust model does not prescribe a specific way of determining trust values. Each system is allowed to provide its own, application specific, trust function.

In this work, our idea is that the trust function could be based on the semantic mismatch computed using an extensional definition of concept interpretation, as described in equation 1.

$$m(P_{\mathcal{D}_i}, P_{\mathcal{D}_r}) = \frac{CEXT(P_{\mathcal{D}_i}) \wedge CEXT(P_{\mathcal{D}_r})}{CEXT(P_{\mathcal{D}_r})} \quad (1)$$

Using an extensional definition we can interpret the resulted value as an account of the frequency of the matching among instances in $P_{\mathcal{D}_i}$ and in $P_{\mathcal{D}_r}$. Then we can use this value to implement process mining techniques, as described in Section IV. The question is now how we can calculate the value of m in a 'Big Data' scenario and how can we use these values in a 'map-reduce' algorithm. Our idea is to distinguish between two different procedures. A first procedure, illustrated in Figure 4, is based on a 'map-reduce' algorithm and it is aimed at processing the integrated data sets to provide real time answers in determining statistic information about data. The values of m are used to count occurrences weighting them according to the frequencies recorded in the second procedure. This second procedure, illustrated in Figure 5, is aimed at evaluating the values m for all the predicates of the common model used in integrating data. This procedure is not required to process data in realtime. Instead, it is intended to apply an inductive classification, aimed at comparing the instances of the different data sets, in order to verify if similar elements belongs to the same class or not. For instance, this can be achieve using extensions of the well-known k-Nearest Neighbor algorithm, specifically designed for working with RDF data [27] [28]. Comparing the instances of a data set $\mathcal{D}_r$ and a data set $\mathcal{D}_i$, we have to find whatever an instance in a give data set is classified under a different class in the reference data set. This can be done computing a distance between the instances in $\mathcal{D}_i$ and classifying them in $\mathcal{D}_r$, according to the classification followed by the k most similar instances in $\mathcal{D}_r$. In other words the result of the k-NN classification provide us with an extensional definition of the intersection among two concepts: $CEXT(P_{\mathcal{D}_i}) \wedge CEXT(P_{\mathcal{D}_r})$.

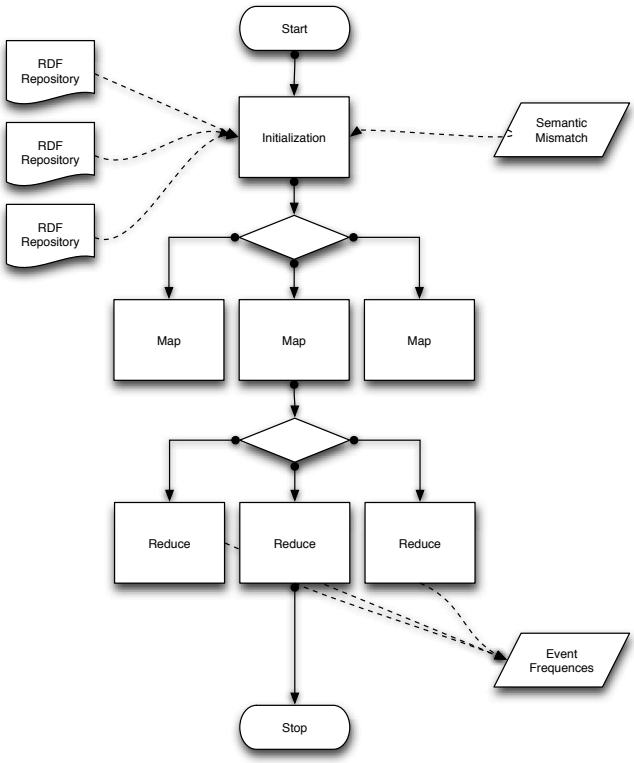

Figure 4. Map Reduce Procedure.

As an example, we suppose to have three RDF data set describing the posting activities of three different online communities, integrated using the SIOC model, as listed in the triples in Figure 6. Suppose to have chosen $D1$ as the $\mathcal{D}_r$ and to have calculated m as reported in Table IV. During the initialization step of a data set $\mathcal{D}_i$ the triples are re-written by simply adding a new triple for each possible interpretation of a predicate, and annotating the triple with the m value. This way, the map reduce algorithm can process

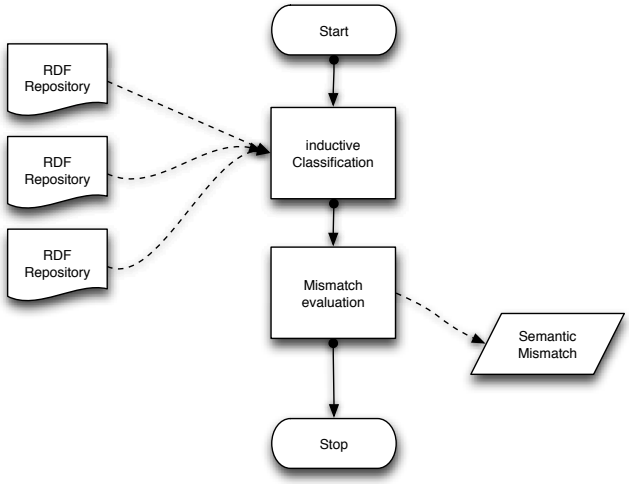

Figure 5. Semantic Mismatch Procedure.

data ordinary. Nevertheless, in the aggregation step for the dependency/frequency tables definition, the frequencies are weighted by considering the m value, as reported in Figure 7. The frequency table generated by using the re-written triples is reported in Table V.

Figure 6. Data described as RDF triples.

```
--D1--
uP create pAAA, pAAA has_content sioc:Text
pAAA has_reply pAA1, pAAA has_view vAA1
uP create pAAB, pAAB has_content sioc:Text
pAAB has_reply pAB1, uP delete pAAB
uP create pAAC, pAAC has_content sioc:Website
pAAC has_reply pAC1, pAAC has_view vAC1
--D2--
uF create pAAD, pAAD has_content sioc:Picture
pAAD has_reply pAD1, pAAD has_reply pAD2
pAAD has_view vAD1
uD create pAAE, pAAE has_content sioc:Article
pAAE has_reply pAE1, pAAE has_reply pAE2
pAAE has_view vAE1
uG create pAAF, pAAF has_content sioc:Picture
pAAF has_reply pAF1, pAAF has_reply pAF2
uG delete pAAF
--D3--
uV create pAAG, pAAG has_content sioc:Text
uV create pAAH, pAAH has_content sioc:Text
pAAH has_reply pAH1, pAAH has_reply pAH2
pAAH has_view vAH1
uV create pAAI, pAAI has_content sioc:Text
uA create pAAG, pAAG has_content sioc:Website
pAAL has_reply pAL1, pAAL has_reply pAL2
pAAL has_view vAL1
uV create pAAM, pAAM has_content sioc:Text
```

VI. Conclusion

'Big Data' techniques are often adopted in cross-organization scenarios for integrating multiple data sources to extract statistics or other latent information. In particular, this work claims that the 'Big Data' challenges are not only related to store and manage the vast volume of data, but also to analyze and extract consistent information from it. Even if these techniques do not require the support of a schema for processing data, a common conceptual model is typically defined to address name resolution. Such an aspect implies that each local source is tasked of applying a semantic lifting procedure for expressing the local data in term of the common model, by potentially introducing semantic heterogeneity in data.

For this reason the semantic lifting problem, among others, is of relevance for developing reliable techniques for processing distributed data. In our dissuasion we introduced an approach specifically tailored for process mining techniques. Our aim was to investigate solutions for handling semantic lifting without compromising the requirements imposed in a 'Big Data' context. Two different procedures were exploited: the first one was aimed at computing the mismatch among the data sources to be integrated, while the second one used mismatch values to extend data to be processed with a traditional map reduce algorithm.

Table IV
Predicate mismatch in different datasets.

m	$Website_{\mathcal{D}_r}$	$Article_{\mathcal{D}_r}$	$Picture_{\mathcal{D}_r}$
$Website_{\mathcal{D}_2}$	0.8	0.2	0
$Article_{\mathcal{D}_2}$	0.2	0.6	0.2
$Picture_{\mathcal{D}_2}$	0	0	1
$Website_{\mathcal{D}_3}$	0.7	0.2	0.1
$Article_{\mathcal{D}_3}$	0.1	0.6	0.3
$Picture_{\mathcal{D}_3}$	0.1	0.1	0.8

Figure 7. RDF triples re-written using the m value.

```
--D1--
uP create pAAA, pAAA has_content sioc:Text
pAAA has_reply pAA1, pAAA has_view vAA1
uP create pAAB, pAAB has_content sioc:Text
pAAB has_reply pAB1, uP delete pAAB
uP create pAAC, pAAC has_content sioc:Website
pAAC has_reply pAC1, pAAC has_view vAC1
--D2--
uF create pAAD, pAAD has_content sioc:Picture
pAAD has_reply pAD1, pAAD has_reply pAD2
pAAD has_view vAD1
uD create pAAE, pAAE has_content sioc:Article 0.6
pAAE has_content sioc:Website 0.2
pAAE has_content sioc:Picture 0.1
pAAE has_reply pAE1, pAAE has_reply pAE2
pAAE has_view vAE1
uG create pAAF, pAAF has_content sioc:Picture
pAAF has_reply pAF1, pAAF has_reply pAF2
uG delete pAAF
--D3--
uV create pAAG, pAAG has_content sioc:Text
uV create pAAH, pAAH has_content sioc:Text
pAAH has_reply pAH1, pAAH has_reply pAH2
pAAH has_view vAH1
uV create pAAI, pAAI has_content sioc:Text
uA create pAAG, pAAG has_content sioc:Website 0.7
pAAG has_content sioc:Article 0.2
pAAG has_content sioc:Picture 0.1
pAAL has_reply pAL1, pAAL has_reply pAL2
pAAL has_view vAL1
uV create pAAM, pAAM has_content sioc:Text
```

Table V
EVENT FREQUENCY.

Dependence m	#	Reply		Delete		View	
		$\#\succ$	$\#\succ^2$	$\#\succ^2$	$\#\succ^3$	$\#\succ^2$	$\#\succ^3$
Create-Text	6	0.5	0.1	0.1	0	0.1	0.1
Create-Website	1.9	1.2	0.5	0	0	0.6	0.5
Create-Picture	2.2	1	1	0	0.4	0	0.5
Create-Article	0.8	1	1	0	0	0	1

In future developments the problem should be framed in a more general theory. For instance considering it from the point of view of the the belief revision problem that studies the problem of integrating new information with previous knowledge [29].

ACKNOWLEDGMENT

This work was partly funded by the Italian Ministry of Economic Development under the Industria 2015 contract - KITE.IT project.

REFERENCES

[1] D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, and R. Rosati, "Conceptual modeling for data integration," in *Conceptual Modeling: Foundations and Applications*. Springer, 2009, pp. 173–197.

[2] P. Hayes and B. McBride, "Rdf semantics," W3C, Tech. Rep. http://www.w3.org/TR/2004/REC-rdf-mt-20040210/, February 2004.

[3] R. Hull, "Managing semantic heterogeneity in databases: a theoretical prospective," in *Proceedings of the sixteenth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems*. ACM, 1997, pp. 51–61.

[4] M. zur Mühlen and R. Shapiro, "Business process analytics," in *Handbook on Business Process Management 2*. Springer, 2010, pp. 137–157.

[5] G. Cugola and A. Margara, "Processing flows of information: From data stream to complex event processing," *ACM Computing Surveys (CSUR)*, vol. 44, no. 3, p. 15, 2012.

[6] V. Borkar, M. J. Carey, and C. Li, "Inside "big data management": ogres, onions, or parfaits?" in *Proceedings of the 15th International Conference on Extending Database Technology*, ser. EDBT '12. New York, NY, USA: ACM, 2012, pp. 3–14.

[7] C. Houy, P. Fettke, P. Loos, W. M. van der Aalst, and J. Krogstie, "Business process management in the large," *Business & Information Systems Engineering*, vol. 3, no. 6, pp. 385–388, 2011.

[8] J. G. Breslin, A. Harth, U. Bojars, and S. Decker, "Towards semantically-interlinked online communities," in *The Semantic Web: Research and Applications*. Springer, 2005, pp. 500–514.

[9] P. Hayes and B. McBride. (2004) Resource description framework (rdf). [Online]. Available: http://www.w3.org/

[10] J. Carroll, C. Bizer, P. Hayes, and P. Stickler, "Named graphs," *Journal of Web Semantics*, vol. 3, no. 3, 2005.

[11] E. Prudhommeaux and A. Seaborne. (2008) Sparql query language for rdf. [Online]. Available: http://www.w3.org/

[12] K. Rohloff, M. Dean, I. Emmons, D. Ryder, and J. Sumner, "An evaluation of triple-store technologies for large data stores," in *On the Move to Meaningful Internet Systems 2007: OTM 2007 Workshops*. Springer, 2007, pp. 1105–1114.

[13] T. Neumann and G. Weikum, "Scalable join processing on very large rdf graphs," in *Proceedings of the 2009 ACM SIGMOD International Conference on Management of data*, ser. SIGMOD '09. New York, NY, USA: ACM, 2009, pp. 627–640. [Online]. Available: http://doi.acm.org/10.1145/1559845.1559911

[14] M. Leida, B. Majeed, M. Colombo, and A. Chu, "Lightweight rdf data model for business processes analysis," *Data-Driven Process Discovery and Analysis, Series: Lecture Notes in Business Information Processing*, vol. 116, 2012.

[15] C. Liu, G. Qi, H. Wang, and Y. Yu, "Large scale fuzzy pd* reasoning using mapreduce," in *Proceedings of the 10th international conference on The semantic web - Volume Part I*, ser. ISWC'11. Berlin, Heidelberg: Springer-Verlag, 2011, pp. 405–420.

[16] J. Urbani, J. Maassen, N. Drost, F. Seinstra, and H. Bal, "Scalable rdf data compression with mapreduce," *Concurrency and Computation: Practice and Experience*, vol. 25, pp. 24–39, 2013.

[17] N. Soule, "Efficient sparql query processing via map-reduce-merge," Boston, MA, USA, 2012.

[18] J. Dean and S. Ghemawat, "Mapreduce: simplified data processing on large clusters." in *in Proceedings of the USENIX Symposium on Operating Systems & Implementation (OSDI)*, 2004, pp. 137–147.

[19] K. Bakshi, "Considerations for big data: Architecture and approach," in *Proceedings of the International Conference on Aerospace*. IEEE Publisher, 2012, pp. 1–7.

[20] J. Buijs, "Mapping data sources to xes in a generic way, master's thesis," Eindhoven, The Netherlands, 2010.

[21] A. D. Nicola, T. D. Mascio, M. Lezoche, and F. Tagliano, "Semantic lifting of business process models," *2012 IEEE 16th International Enterprise Distributed Object Computing Conference Workshops*, vol. 0, pp. 120–126, 2008.

[22] W. Van Der Aalst and K. Van Hee, *Workflow management: models, methods, and systems*. MIT press, 2004.

[23] P. Smyth and R. M. Goodman, "Rule induction using information theory," *Knowledge discovery in databases*, vol. 1991, 1991.

[24] W. van der Aalst, "Process mining: Discovering and improving spaghetti and lasagna processes," Keynote Lecture, IEEE Symposium Series on Computational Intelligence (SSCI 2011)/IEEE Symposium on Computational Intelligence and Data Mining (CIDM 2011), April 2011.

[25] M. Hert, G. Reif, and H. C. Gall, "A comparison of rdb-to-rdf mapping languages," in *Proceedings of the 7th International Conference on Semantic Systems*, ser. I-Semantics '11. New York, NY, USA: ACM, 2011, pp. 25–32. [Online]. Available: http://doi.acm.org/10.1145/2063518.2063522

[26] K. Reddy and P. S. Kumar, "Efficient trust-based approximate sparql querying of the web of linked data," in *Uncertainty Reasoning for the Semantic Web II*, ser. Lecture Notes in Computer Science, F. Bobillo, P. Costa, C. dAmato, N. Fanizzi, K. Laskey, K. Laskey, T. Lukasiewicz, M. Nickles, and M. Pool, Eds. Springer Berlin Heidelberg, 2013, vol. 7123, pp. 315–330.

[27] C. d'Amato, N. Fanizzi, and F. Esposito, "Query answering and ontology population: An inductive approach," in *The Semantic Web: Research and Applications*. Springer, 2008, pp. 288–302.

[28] C. d'Amato, F. Esposito, N. Fanizzi, B. Fazzinga, G. Gottlob, and T. Lukasiewicz, "Inductive reasoning and semantic web search," in *Proceedings of the 2010 ACM Symposium on Applied Computing*. ACM, 2010, pp. 1446–1447.

[29] O. Papini, "Knowledge-base revision," *Knowl. Eng. Rev.*, vol. 15, no. 4, pp. 339–370, Dec. 2000.