Argumentation-based Explanation of Linked Data Fusion

Fatiha Saïs, Rallou Thomopoulos, Anderson Carlos Ferreira da Silva

Argumentation-based Explanation of Linked Data Fusion

1st Fatiha Saïs
LRI, Paris-Sud University
CNRS 8623, Paris-Saclay University
Orsay, France
fatiha.sais@lri.fr

2nd Rallou Thomopoulos
INRA (UMR IATE) & INRIA GraphIK
2 place Viala
Montpellier, France
rallou.thomopoulos@inra.fr

3rd Anderson Carlos Ferreira Da Silva
LRI, Paris-Sud University
CNRS 8623, Paris-Saclay University
Orsay, France
anderson-carlos.ferreira-da-silva@u-psud.fr

***Abstract*—This paper deals with redundant data, previously identified by a data linkage step. The question considered is: how to propose a unique representation by merging the identified "duplicates"? More specifically, how to decide, for each data property, which value will be chosen among those describing the "duplicates"? What method should be adopted in order to be able to trace and explain the result to the user in an understandable form? The proposed approach relies both on multi-criteria decision and argumentation, combined with the computation of a quality score.**

***Index Terms*—Data Fusion, Explanation, Linked Data, Data Quality**

I. Introduction

Nowadays, the *Web of Data* is one of the most important fields of the *Semantic Web*. It is increasingly used and acknowledged in a variety of application areas. It can be described as a structured way to store data and to interconnect them with meaningful correspondences. These connections among data objects are extremely useful, as they allow different and often heterogeneous data to be explored and queried by applications, thus expanding the data space.

The *Linked Open Data* (LOD) project [3], conceived in 2007, is a fundamental initiative in this direction. It supports the aggregation and interconnection of numerous data that are already available on the Web. Rapidly growing, it now contains over 149 billion RDF (Resource Description Framework)[1] triples linked with over 500 million RDF links[2], composing an enormous area of knowledge.

Within the setting of LOD, it is often the case that objects, possibly coming from different data sources, represent the same real-world entity. In order to maintain the usability of the LOD, it is critical to detect this kind of relations between objects and to attempt to obtain one single object describing the real-world entity. Naturally, data are constantly evolving, new data are generated and creating links between objects becomes more and more complex. It is thus necessary to use automatic procedures to this end.

Data linking, also known as *entity resolution or data reconciliation*, is the process where two object descriptions are examined in order to determine whether they refer to the same real-world entity, and if so, to link them together through `owl:sameAs` predicates. Many approaches have been proposed to link RDF data, some are logic based like [11], other are similarity or probability based such as [12], [15]. For more details on data linking approaches, see surveys in [7].

Then, *data fusion* encompasses the effort to acquire a single homogenized object by merging the conflicting information of the linked individual objects. The objects linked by the `owl:sameAs` may contain different, conflicting or inconsistent values in their properties. For each property, the conflict on its different values must be compromised and the most appropriate value must be chosen. The task of data fusion consists in merging the individual representations by resolving these conflicts and generating one single final object containing the whole set of information describing the objects. Data fusion is an essential step towards avoiding redundancy, grouping together the best quality information and giving consistent answers to the users, in the linked data environment.

Some works has investigated the problem of data fusion in the field of relational databases (see [4] for a state of the art). In this article, we are interested in the context of the Web of data and we study the problem of RDF data fusion, which is different from the relational framework, because characterized by the intrinsic flexibility of the model allowing the multi-valuation of the properties, the open world assumption and the possibility of having several ontologies (schemas) describing the data. In this setting some works such as [8], [10], [13], [14] have addressed the problem of RDF data fusion. However none of these approaches allows a good user understanding of the data fusion results. In this paper, we focus on the explanation aspect, that aims at keeping track and giving back the reasons that led to a fusion result.

The research question that we attempt to address in the paper is how argumentation theory may help in creating explanations for data fusion process. Indeed, we choose argumentation theory [6] because it is a suitable tool when evaluating the arguments in favour of and arguments against a claim/decision in a decision process or in expert dialogues. Argumentation is already used for the explanation of decision processes [5] in the food science domain and also for explaining the identity link invalidation [2]. As a data fusion method we base our

[1] https://www.w3.org/RDF/
[2] http://stats.lod2.eu/

study on the multi-criteria method that we presented in [9]. By exploiting the criteria such as value frequency, data source reliability, semantic relations, our explanation approach is able to generate arguments that are in favour or against a value to be the best value for given property (e.g. a museum address). These arguments are on the one hand, exploited to determine or to strengthen the fact that a value should be the right value among conflicting values of a given property. On the other hand, these arguments are used to provide explanations to users on the obtained data fusion results.

In section II we give an overview of RDF data fusion existing works. Section III presents the materials needed to the definition of our method. In section IV we present our multi-criteria data fusion method and in V we show how we use the argumentation framework to generate explanations for data fusion results. Finally, we give some very first experiments of real datasets in section VI and some concluding remarks and future works in section VII.

II. Related Work

Research on the data fusion problem has begun over two decades ago in the field of relational databases. The survey of Bleiholder and Naumann [4] outlines the state of the art in this direction. However, as we examine the data fusion from the RDF point of view, we notice that the specificities of RDF mechanisms cannot be reflected in solutions offered by relational databases experts. Three main approaches have been proposed for data fusion in RDF. These different approaches attempt to evaluate the quality of each value, by taking into account various measures based on the value itself and/or data source metadata.

The approach proposed by Saïs and Thomopoulos in 2008 [13] consists in establishing a confidence degree for each value, by calculating a number of quality criteria for each property, and combining them. Then, the values with higher confidence degrees are ranked higher. The result for a property is a fuzzy set of all the possible values, thus the uncertainty is captured. In a more recent work [14], the uncertainty of data fusion results is modelled trough possibility theory. Flouris et al. in 2012 [8] propose a similar approach. A combination of quality metrics is used as well, to select the most appropriate value. Here, a single value is chosen the one with the highest quality score. In Mendes et al. 2012 [10], the framework Sieve is introduced as part of the Linked Data Integration Framework (LDIF), to deal with data quality assessment and fusion. Concerning the fusion phase, Sieve handles conflicts with three strategies, based on the idea described in [4]: (i) *conflict-ignoring* strategies, where conflicts are left to be resolved by the users; (ii) *conflict-avoiding* strategies, where one decision is uniformly applied to all attributes (example: always trust a specific source); and (iii) *conflict-resolution* strategies, where one of the existing values is *decided* or a new value is *mediated* (e.g., the average, or the maximum of all the given values).

The term *instance based* refers to strategies where the decision is made according to properties of the value itself (e.g., frequency) as opposed to the term *source metadata based*, where the choice is based on information of the data sources (reliability, freshness, etc.) The term *deciding* declares that the final value will always come from the list of the existing values, as opposed to the *mediating* strategies that can produce new values (e.g., average for a person's age). It is worth mentioning that Sieve is more a framework than a method per se, and thus, as we will see below, our proposition can be positioned with respect to it.

After examining the related works, we pointed out their main deficiencies concerning the problem we tackle.

- *Treating all values uniformly*. This means that none of the three methods applies different criteria according to the values nature. We believe that specialized measures and scores (e.g., similarity scores for symbolic values, ranges for numerical ones, constraints for dates, etc.), could offer an advantage to the approaches.
- *Not using the ontology knowledge*. Using RDF and OWL data offers the opportunity to benefit the ontology semantics and knowledge, such as relations between properties. As we will see, important hints can be extracted by these relations, that can indicate whether a value is acceptable or not.
- *Not catering for multi-valued properties*. A fairly common case for RDF attributes is to be multi-valued, that is, to accept more than one values. We did not find any satisfactory solution for this case.
- *Heavy user involvement*. We also find that the involvement of the users in Flouris et al., where they have to choose a preferred source, as well as in Mendes et al., where they have several levels of involvement, although flexible, is not easily applicable for large scale data integrations.
- *Not offering explanations for fusion decisions*. An admittedly useful concept is to keep track of the reasons behind each fusion decision, in order to explain to the users, often, unexpected results. A first step in this direction has been accomplished in Saïs and Thomopoulos in 2008 [13], by annotating each value with its confidence degree in the resulting dataset. However, the confidence degree alone cannot provide sufficient explanations to the user regarding how this degree was computed.

III. Preliminaries

A. Knowledge Bases

We consider knowledge bases that are defined by an ontology $\mathcal{O}$ represented in OWL[3] and facts $\mathcal{F}$ represented as a collection of RDF triples.

Definition 1. (**Knowledge base**) A knowledge base $\mathcal{B}$ is defined by a couple $(\mathcal{O}, \mathcal{F})$ where:

– $\mathcal{O} = (\mathcal{C}, \mathcal{P}, \mathcal{A})$ represents the conceptual part of the knowledge base defined by a set of classes $\mathcal{C}$, a set of

[3]https://www.w3.org/OWL/

properties (*owl*:*DataTypeProperty* and *owl*:*ObjectProperty*) $\mathcal{P}$ and a set of axioms $\mathcal{A}$ that represents relations such as the subsumption and disjunction for classes, or (inverse) functionality for properties.

– $\mathcal{F}$ is a collection of RDF triples $< s,\ p,\ o >$, of a subject s that is a URI, a property $p \in \mathcal{P}$ that is also a URI, and an object o that can be either a URI or a Literal. We consider the subjects and the objects as instances associated to one or more classes of $\mathcal{O}$ by the *rdf:type* property. We do not consider blank nodes in this work. We denote the set of values of a given property p by V_p = $\{x| < \_, p, x >\in \mathcal{F}$ or $< x, p, \_ >\in \mathcal{F}\}$.

In the sequel we will refer by the term *instances* the URIs appearing as a subject or an object of the triples in $\mathcal{F}$ (except the objects of *rdf:type* triples).

B. Identity links

Definition 2. (Identity link). *An identity link is a relationship id that can be declared for every pair of instances referring to the same real world entity. It can be expressed by predicates such as owl:sameAs predicate whose semantics follows the classical definition of identity relation. It requires the following identity properties:*

1) *reflexivity* $(x = x)$,
2) *symmetry,* $owl{:}sameAs(i_1, i_2) \Rightarrow owl{:}sameAs(i_2, i_1)$.
3) *transitivity,* $owl{:}sameAs(i_1, i_2) \wedge owl{:}sameAs(i_2, i_3) \Rightarrow owl{:}sameAs(i_1, i_3)$,
4) *property sharing, meaning that all the properties of the related instances have the same values,* $owl{:}sameAs(i_1, i_2) \wedge p(i_1, v) \Rightarrow p(i_2, v)$).

We note that there exist other predicates such as *skos*:*exactMatch* or built-in predicates (e.g. reconciled, identical) that can be used to represent the identity relationship between instances. In this work we consider identity predicates that fulfil the four identity properties exhibited in Definition 2. We denote the set of possible identity predicates by $ID_{predicates}$.

C. Identity link set

An identity link set is a set of RDF triples expressing an identity relation between two resources. According to the W3C[4] a linkset is a set of RDF triples where all subjects are in one dataset and all objects are in another dataset. RDF links often have the *owl:sameAs* predicate, but any other property could occur as the predicate of RDF links as well.

In this work we consider a more general definition of a link set by considering the case where the subjects and the objects involved in the identity link may either be in the same knowledge base (dataset) or in different knowledge bases. We restrict the set of identity predicates to those fulfilling the identity properties (see Definition 2).

[4]https://www.w3.org/TR/void/#linkset

Definition 3. (Link Set). *A link set ILS is a set of RDF triples where for all triples $t_i =< s_i, p_i, o_i >\in L$ s_i and o_i are URIs referring to instances represented in one or different knowledge bases and $p_i \in ID_{predicates}$ an identity relation.*

D. Data Fusion Problem

The problem of data fusion is the task attempting to compute a fused knowledge base that is *complete*, *correct* and *concise* as most as possible, for a given set of n knowledge bases and a set of identity links between the instances of the n knowledge bases. We consider a set of instances $I = \{i_1, i_2, \ldots, i_n\}$ that are pairwise linked by an identity relation (e.g. *owl:sameAs*, a built-in predicate *reconciled*) and that represent an equivalence class which can be obtained from the link set ILS. Each instance is described by a set of properties $P = \{p_1, p_2, \ldots, p_m\}$ declared in a common ontology or in several aligned ontologies (using ontology alignment tools). This set of properties can be obtained by computing a transitive closure on $\mathcal{M}$ and by keeping a representative property for each property equivalence class.

In the sequel the set I will be referred to by the term *equivalence class* and the union of the values of a property $p \in P$ for all the instances $i \in I$ is referred to as *set of possible values*.

E. Illustrative Example

We first consider a set of facts representing three different descriptions of the same book $b1$, $b2$ and $b3$. Each book is described by the whole following properties (description from schema.org): {*title, nbPages, author, contributor, publisher, dateCreated, dateModified, datePublished, keywords*}. We assume that the property mappings have already been applied and led to a uniform property set. We assume that a data linking tool has been previously applied and returned the following result:

$<$ b1 $>$ owl : sameAs $<$ b2 $>$. $<$ b1 $>$ owl : sameAs $<$ b3 $>$.
$<$ b2 $>$ owl : sameAs $<$ b3 $>$.

In Figure 2, we show examples of two incompatibility rules between the values of some properties and a implausibility rule for the property *nbPages*.

IV. Multi-Criteria Data Fusion Method

After introducing the terminology and symbolism used in this work, in this section, we present our data fusion method.

A. Multi-criteria Decision Problem

We consider the task of selecting the right value for each property p describing the instances of an given equality set I as a multi-criteria decision making problem. The inputs of the problem, namely, the set of the considered options in one side, and the set of criteria that are used to discriminate between the different options in another side, are as follows:

1) the set of considered options is the set of values of p of I ;

@prefix ob : < https : //schema.org/Book > .

uri	ob:title	ob: nbPages	ob:author	ob:contributor	ob:publisher
b1	A Semantic Web Primer	238	Grigoris Antoniou	Paul Groth Frank V. Harmelen Rinke Hoekstra	The MIT Press (MA)
b2	A Semantic Web Primer	0	G. Antoniou	P. Groth F. V. Harmelen R. Hoekstra	MIT Press MA
b3	A Semantic Web Primer, second edition (cooperative information Systems Series	288	Grigoris Antoniou	Paul Groth Frank Van Harmelen Rinke Hoekstra	MIT Press Massachusetts

uri	ob:dateCreated	ob:dateModified	ob:datePublished	ob:keywords
b1	12/07/2007	01/05/2008	03/01/2008	Computer Science Knowledge representation Semantic Web
b2	December 7th 2007	April 30th 2004	March 1st 2008	Artificial Intelligence Description Logic Semantic Web
b3	December 2007	January 2008	March 2008	Semantic Web AI Knowledge representation & reasoning

Fig. 1. Examples of three descriptions of the same book

Implausibility rule:	R1: nbPages ≤ 0
incompatibility rules :	R2 : dateModified $\geq$ datePublished
	R3 : dateModified $\leq$ dateCreated

Fig. 2. Examples of implausibility and incompatibility rules

2) the set of considered criteria includes: value plausibility, precision, compatibility, homogeneity, frequency, synonymy relation, source frailness and reliability.

In the following section, we give details on the set of criteria and their application.

B. Multi-criteria Data Fusion Methodology

Our data fusion approach is based on a set of criteria and proceeds according the following steps:

Implausible values filtering. A filtering step is performed in order to determine and filter-out the property values that do not follow some of known domain constraints and some property typing constraints. For instance, if the property *pages* is typed as "*xsd:nonNegativeInteger*" then a negative value of it should be considered as implausible.

More-precise relation. This step consists in pairwise comparing the set of values of a given property in a given equality set and determines if there is a more-precise relation. To do so, we exploit: (i) syntactic comparisons between strings, (ii) subsumption relations and (ii) mereology (*part-of*) relation. For the two former ones, we reasonably assume that we may assign to some properties whose values can be hierarchically organised, available hierarchies such as geographical classifications and concept hierarchies. Thus, the hierarchical structure is used to check whether a value is more precise than an other. For example, the value *"Knowledge Representation"* is more precise than *"Computer Science"* and *"Massachusetts"* is more precise than *"USA"*.

Synonymy relations. This step aims at determining synonymy relations between pairs of values of a given property in a given equality set. To do this, we use available dictionaries like synsets of WordNet [5]. For example the value *"AI"* is synonym of *"Artificial Intelligence"*.

Incompatibility relations. This step aims at identifying the property values that violate the compatibility rules provided by a domain expert. These rules which may involve one or several properties. For example, the value *03/01/2008* of the property *ob:datePublished* is incompatible with the value *01/05/2008* of the property *ob:dateModified*?

Quality score computation. This step aims at computing the quality score for each plausible value. This score is obtained using an aggregation function of measured criteria concerning the value itself (homogeneity, frequency, precision) and other criteria that are related to the quality of the original data source (freshness and reliability). For more details on these criteria see [13].

Let $qs(v)$ be the function that computes a quality score for a value v. Let S be the sources from which v is originated. The function $qs(v)$ is computed as follows:

$$qs(v) = \frac{Fresh(S) + Rel(S) + Freq(v) + Homo(v) + |isMPT(v)|}{4 + |isMPT(v)|}$$

where $Fresh(S)$ is the freshness of the source S, $Rel(S)$ is the reliability degree of the source S, $Freq(v)$ is the frequency of the value v in the all the knowledge bases, $Homo(v)$ is the

[5]https://wordnet.princeton.edu/

homogeneity of v in the set of possible values of the property and $isMPT(v)$ represents the set of values among the possible ones that are less precise than v.

V. Explanation of fusion decisions

A. Argumentation to explain a decision

Argumentation [6] turns out to be a relevant tool when the advantages and disadvantages of a claim or a decision must be evaluated on the basis of available knowledge. Very few studies discuss the interest of argumentation theory for decision; since the two fields have historically been studied separately with different objectives. [1] proposes a two-step process: (i) evaluation of the set of arguments built in favour of or against the different options, (ii) from the arguments accepted in step 1, ranking of options by the choice of a preference-based aggregation method. Our proposal has the particularity of using arguments as a " quality tool", for the traceability and the explanation of a decision. We introduce for this the following definition of explanation decision system.

Definition (**Explanation Decision System**.) An explanation decision system is a tuple $\langle D, G, A\rangle$ where:

- D is the set of options;
- G is the set of criteria. To each criteria g, a domain of value $V(g)$ is assigned;
- A is the set of arguments. Each $a \in A$ is defined by a tuple $\langle d_a, type_a, g_a, v_a\rangle$, where:
  - $d_a \in D$ is the option considered by the argument a;
  - $type_a$ is a boolean indicating the type of the argument a which can either *in favour of* ($type_a = true$) or *against* ($type_a = false$) the option d_a;
  - $g_a \in G$ is the criteria on which the argument a is relying to express oneself in favour of or against the option d_a;
  - $v_a \in V(g_a)$ is the value taken by g_a regarding to the option d_a.

From the explanation decision system above, two functions allow to provide the **explanation of the choice or elimination** of an option:

- $\forall d \in D,\ F_+(d) = \{a \in A | type_a = true\ \wedge\ d_a = d\}$ *assigns to each option* d *the set of arguments that are in favour of* d*;*
- $\forall d \in D,\ F_-(d) = \{a \in A | type_a = false\ \wedge\ d_a = d\}$ *assigns to each option* d *the set of arguments that are against* d*.*

B. Arguments Construction: Process and Example

We consider the different steps of the fusion method presented in section IV-B. We consider a property p with its possible set of values $\mathcal{V}$.

Implausible values. When and implausible value v is detected for a property p, an argument against v is constructed. The criterion on which this argument is relying is the implausibility, which is obtained from the violation of some domain and/or property typing constraints.

Example 1 *From the implausibility rule* $R1$ *given in figure 2 and from the data of figure 1 the value* 0 *for the property* nbPages *is implausible. This value is rejected and the following argument* a_1 *is built against this value:* $a_1 = \langle 0, false, plausibility, \neg R1\rangle$.

Precision relation. When a more-precise relation is detected between two values v and v' of the property p, two arguments are built, one in favour of the more precise option (value), considered as being more informative, another argument against the less precise option, is built. The criterion on which these arguments are based is precision.

Example 2. *For the property* dateCreated *of the figure 1, the value "December 7th 2007" is identified as more precise than "December 2007". The following arguments are constructed:*

$a_2 =$
$\langle$"$December\ 7th\ 2007''$, $true$, $precision$, $\succeq$"$December\ 2007''\rangle$.
$a_3 =$
$\langle$"$December\ 2007''$, $false$, $precision$, $\preceq$"$December\ 7th\ 2007''\rangle$.

Synonymy relation. Detecting synonymy relations does not by itself allow eliminating or selecting one or the other of synonymous values. It requires additional information, either to choose one of the synonyms based on its relevance, or to allow a set of values for the property p, which would justify keeping the two synonyms. However the detection of the synonymy accredits the two values, by establishing that they are not aberrant since they are acceptable variants having the same meaning. For this reason, two arguments are constructed, in favour of both options.

Example 3 *For the property* dateCreated *of the figure 1, the values "December 7th 2007" and "12/07/2007" are identified as synonyms. The following arguments are then built:*

$a_4 =$
$\langle$"$December\ 7th\ 2007''$, $true$, $synonymy$, $= 12/07/2007\rangle$.
$a_5 =$
$\langle 12/07/2007, true, synonymy, =$ "$December\ 7th\ 2007''\rangle$.

Incompatibility Relation. The detection of an incompatibility relation between two values $v1$ and $v2$, for two distinct properties $p1$ and $p2$, introduces a doubt about these two values. As a result, two arguments are generated, against the two options. However, the context and other arguments in favour of or against $v1$ and $v2$ can identify which of the two values is erroneous, or both are erroneous.

Example 4. In the description of $b2$ (Figure 1), the of the properties *dateCreated* and *dateModified* violate the rule $R3$ (Figure 2). Two arguments are generated.

For the property *dateCreated*:

$a_6 = \langle$"$December\ 7th\ 2007''$, $false$, $compatibility$, $\neg R3\rangle$.

For the property *dateModified*:

$a_7 = \langle \text{“}April 30th 2004'', false, compatibility, \neg R3 \rangle$.

Further review (other arguments in favour of or against each value, including $a2$ and $a4$) may allow to conclude that the erroneous value is the value if the property *dateModified* (“April 30th 2004”) and thus eliminate the argument $a6$.

Quality Score. At this stage, the arguments built in the previous steps may conclude on the value to choose for a property p in the fused data. For example, in Figure 1, the precision arguments may be exploited to choose the best value for the properties *author* and *contributor*, and the compatibility arguments may be used to choose the right value for the property *dateModified*. The quality score, computed for any plausible value, allows to rank the property values and continues the fusion process for the remaining properties. The computation of this score, which is not detailed here, is also accompanied by the construction of the arguments, as well as sub-arguments (for each criterion involved in the score).

VI. First Experiments

The data fusion tool has been developed in Java (JDK 1.8.0) within the Netbeans IDE 8.2 Platform.

Through the interface, the user has the possibility to specify several parameters such as the serialisation format of the data source (e.g., N3, Turtle, RDF/XML), the set of ontology mappings, the reliability and the last update date od the data sources.

We run our first experiments on Yago[6] and DBpedia[7] datasets: two datasets that are automatically extracted from Wikipedia using different techniques and different ways to structure the ontology. We considered a set of given equivalence property mappings and a set of *owl:sameAs* links obtained by applying a linking tool. We run our data fusion tool by considering the data separated by class, e.g. Museum, Book, Artist, Organisation, etc. For example the performing of the tool on the class Museum, which has URIs instances in Yago and in DBpedia, 586 *owl:sameAs* links and 7 common properties has led to 489 equivalence class. There size varies from 2 to 4 URIs. The fusion task was achieved in 2 seconds.

VII. Conclusion

This article introduces an argumentation based approach for data fusion explanation, allowing the user to understand the obtained results. We have introduced an explanation decision-making framework, a quality-based tool aimed at tracing and giving back the reasons that led to the selection or elimination of a value in the fused data. Therefore, we relied on an approach combining multi-criteria decision making and argumentation. The different arguments in favour of and against a value are generated by exploiting the knowledge used in the decision-making process. They include precision relations between values, violation of property typing constraints, knowledge about compatibility between values of different properties, but also qualitative and quantitative information about values (e.g., homogeneity, frequency) and metadata on the original sources (e.g., freshness, reliability of sources). We plan to deeper study different directions, and in particular the complex case of multivalued data. Another perspective concerns the order of application of the rules used, and therefore the order of precedence of the arguments that are constructed. This generic approach is not specific to the fusion case but can be adapted and applied to other decision making contexts.

[6]www.yago-knowledge.org/

[7]http://wiki.dbpedia.org/

References

[1] Leila Amgoud and Henri Prade. Using arguments for making and explaining decisions. *Artif. Intell.*, 173(3-4):413–436, 2009.

[2] Abdallah Arioua, Madalina Croitoru, Laura Papaleo, Nathalie Pernelle, and Swan Rocher. On the explanation of sameas statements using argumentation. In *Scalable Uncertainty Management - 10th International Conference, SUM 2016, Nice, France, September 21-23, 2016, Proceedings*, pages 51–66, 2016.

[3] Sören Auer, Volha Bryl, and Sebastian Tramp, editors. *Linked Open Data - Creating Knowledge Out of Interlinked Data - Results of the LOD2 Project*, volume 8661 of *Lecture Notes in Computer Science*. Springer, 2014.

[4] Jens Bleiholder and Felix Naumann. Data fusion. *ACM Comput. Surv.*, 41(1):1:1–1:41, January 2009.

[5] Jean-Rémi Bourguet, Rallou Thomopoulos, Marie-Laure Mugnier, and Joël Abécassis. An artificial intelligence-based approach to deal with argumentation applied to food quality in a public health policy. *Expert Syst. Appl.*, 40(11):4539–4546, 2013.

[6] P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77:321–357, 1995.

[7] Alfio Ferrara, Andriy Nikolov, and François Scharffe. Data linking. *J. Web Sem.*, 23:1, 2013.

[8] Giorgos Flouris, Yannis Roussakis andMaria Poveda-Villalon andPablo N. Mendes, and Irini Fundulaki. Using provenance for quality assessment and repair in linked open data. In *In Proceedings of the 2nd Joint Workshop on Knowledge Evolution and Ontology Dynamics (EvoDyn-12)*, 2012.

[9] Ioanna Giannopoulou, Fatiha Saïs, and Rallou Thomopoulos. Linked data annotation and fusion driven by data quality evaluation. In *15èmes Journées Francophones Extraction et Gestion des Connaissances, EGC 2015, 27-30 Janvier 2015, Luxembourg*, pages 257–262, 2015.

[10] Pablo N. Mendes, Hannes Mühleisen, and Christian Bizer. Sieve: linked data quality assessment and fusion. In *Proceedings of the 2012 Joint EDBT/ICDT Workshops, Berlin, Germany, March 30, 2012*, pages 116–123, 2012.

[11] Fatiha Saïs, Nathalie Pernelle, and Marie-Christine Rousset. L2R: A logical method for reference reconciliation. In *Proceedings of the Twenty-Second AAAI Conference on Artificial Intelligence, July 22-26, 2007, Vancouver, British Columbia, Canada*, pages 329–334, 2007.

[12] Fatiha Saïs, Nathalie Pernelle, and Marie-Christine Rousset. Combining a logical and a numerical method for data reconciliation. *J. Data Semantics*, 12:66–94, 2009.

[13] Fatiha Saïs and Rallou Thomopoulos. Reference fusion and flexible querying. In *Proceedings of On the Move to Meaningful Internet Systems: OTM 2008 Confederated International Conferences, CoopIS, DOA, GADA, IS, and ODBASE 2008, Part II*, pages 1541–1549, 2008.

[14] Fatiha Saïs, Rallou Thomopoulos, and Sébastien Destercke. Ontology-driven possibilistic reference fusion. In *Proceedings of On the Move to Meaningful Internet Systems, OTM 2010 - Confederated International Conferences: CoopIS, IS, DOA and ODBASE, Part II*, pages 1079–1096, 2010.

[15] Fabian M. Suchanek, Serge Abiteboul, and Pierre Senellart. PARIS: probabilistic alignment of relations, instances, and schema. *PVLDB*, 5(3):157–168, 2011.