



On Link Validity and Entity Resolution

Léa Guizol, Madalina Croitoru, Michel Leclère

► To cite this version:

Léa Guizol, Madalina Croitoru, Michel Leclère. On Link Validity and Entity Resolution. RR-11010, 2012, pp.17. lirmm-00647284

HAL Id: lirmm-00647284

<https://hal-lirmm.ccsd.cnrs.fr/lirmm-00647284>

Submitted on 3 Jan 2012

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

On Link Validity and entity resolution

Research report RR-11010

Léa Guizol, Madalina Croitoru, Michel Leclère

LIRMM (University of Montpellier II & CNRS), INRIA Sophia-Antipolis, France

Abstract. The Entity Resolution problem has been widely addressed in the literature. In its simplest version, the problem takes as input a knowledge base composed of records describing real world entities and outputs the sets of records judged to correspond to the same real world entity. More elaborated versions take into account links amongst records representing relationships between the entities which represent. However, none of the approaches in the literature question the validity of certain links between records. In this paper we highlight this new aspect of “link validity” in knowledge bases and show how Entity Resolution approaches should take this aspect into consideration.

1 Introduction

Knowledge base systems (KBs) allow to store and query an abstract model of the real world using a representation and reasoning language based on formal logic. One of the main problem when managing such a system is to ensure that the users of the system share the same “representation/interpretation” relationship between the conceptual primitives of the language and their corresponding notions in the real world. The development of domain ontologies which fix the vocabulary for classes and properties and specify, by axioms (some specific formulas), their semantics establishes a first solution to this problem. For individual entities, this solution is not applicable. Indeed, we have to continually reference new individuals, and the number of individual references to manage can reach several thousand (or million) individuals. To tackle this problem, a record is associated with each individual reference that specifies the characteristics of the referred individual entity. At least, this record contains, generally a name attribute which indicates the names which are used in the real world to designate the corresponding entity and a type attribute which indicates its class in addition to the reference which identifies the record. For instance, a record corresponding to a literary text contains the “work” class as type and a title as name.

Often, users of the knowledge base own very little information about an individual entity and this information is rather contextual. For instance, when a user inserts a new book in a bibliographic base, often the only information (s)he has about author, the author’s name on the cover. Unfortunately names don’t identify a real world entity, neither its corresponding record. This is due to abusive use of abbreviations, variants, homonyms, etc. As a matter of consequence, many records (and thus references) in the knowledge base represent the same individual entity (real world).

For this purpose, the Entity Resolution problem has been widely addressed in the literature. The problem takes as input a knowledge base composed of records (identified by references which represent a real world entity) usually implemented using relational inspired models such as relational databases or data warehouses (semi-structured approaches as in graphs, triple representations etc. have also been investigated). The output of an Entity Resolution problem is, in this case, the knowledge base references corresponding to the same real world entity. Some approaches go further and propose repairs to the knowledge base (see for instance [2] for merge approaches).

Since the paper of Newcombe [9], there have been hundreds of approaches addressing the problem [6]. Ironically, the Entity Resolution (ER) problem is encountered under several different names: Record Linkage (e.g. [13]), Record Matching (e.g. [7]), Reference Reconciliation (e.g. [11]), Entity Resolution (e.g. [3]), Entity Matching (e.g. [4]), Name Disambiguation (e.g. [12]), Data Interlinking (e.g. [14]).

As a common denominator of the ER approaches above, the KB references to be resolved are either (1) linked with other knowledge base references considered to be correct (from an ER view point) or, (2) linked amongst themselves. Special approaches dedicated to the first case have already been proposed in the literature (see for instance [5]). Similarly, when the references to be resolved are interlinked (cf. second case) special propagation techniques were developed [6]. Getoor and colleagues [3], investigate link based Entity Resolution. Partially using techniques above, the ER problem is translated into the link mining problem between references.

However, all of these techniques rely on the fundamental assumption that the references within the knowledge base are correctly linked amongst themselves. In this paper we investigate the link validation problem (whether references are correctly linked amongst themselves). This is different from the Entity Reference problem since we do not aim to say two references point to the same entity. What we aim to do, is to decide if one reference correctly points to another reference (under certain assumptions). In the light of the aspects mentioned above, this paper:

- ✓ Highlights the problem of knowledge base link validity
- ✓ Proposes a framework allowing for both:
 - ✓ Checking the knowledge base link validity and,
 - ✓ Repairing erroneous links

To this end, we:

- Address the link validation problem using Entity Resolution on enriched entities.
- Lay down current and future work directions improving the accuracy of the previous point where criteria used for ER are given using a preference order.

2 Documentary Bases Link Validation

As a motivating example, we present a real world example from a documentary base manipulation use case within a joint project with ABES. Since 2001, ABES (French Bibliographic Agency for Higher Education) has been managing SUDOC¹ (University

¹ <http://en.abes.fr/Sudoc/The-Sudoc-catalog>

System of Documentation), a French collective catalog containing over 10 million bibliographic records. In addition to *bibliographic records* that describe the documents of the collections of the French university and higher education and research libraries, it contains nearly 2.4 million *authority records* that describe individual entities (or named entities) useful for the description of documents (persons, families, corporate bodies, events etc.). Bibliographic records contain *links* to authority records that identify individuals with a specific role (author, editor etc.) wrt the document described.

A typical entry of a book in SUDOC takes place as follows. The librarian has a new book to enter in the system. (S)he types in and enters the title of the book ("The Cocktail Party"), the ISBN, the number of pages and so forth. Then the librarian needs to indicate the authors of the book from the author names (s)he sees on the cover of the book (in our case "T.S. Eliot"). (S)he cannot directly type this in the author field of his entry. What (s)he needs to do it to search in the SUDOC base the person authority references that could be a good candidate for each author of the book (based on the name and surname), decide if one of the authors existing is suitable, and make the link from the bibliographic record to the authority record by indicating in the author field the reference of the selected authority. If none of the authors in the base is suitable, then the librarian will create a new authority record in the system and link the book to this new record. To continue the example of the "Cocktail Party", the librarian searches the SUDOC for "T. S. Eliot". The system, let's assume, will give three candidates: "Eliot", "T. Eliot (1958-....)" and "G. Eliot". The librarian, at this stage, can make several mistakes. Either choose "Eliot" to be linked from the "Cocktail Party" (and not consider suspicious the fact that this "Eliot" in question published both novels and mathematics books). Either choose "T. Eliot (1958-....)" (surprised that the author has been published so young, the book dating from 1974). Either, mistakenly (it is a human domain expert, after all), choose "G. Eliot". The librarian can also, if in doubt, choose to create his / her own "T.S. Elliot" person authority record. The lack of distinguishing characteristics in the authority records and the lack of knowledge about the identity of the book's author imply that the librarian's decision is mainly based of consultation of previous bibliographic records linked to each candidates (s)he considers. So any linkage error will entail new linkage errors.

When validating a knowledge base we need to distinguish between information which we are sure of, and information which is unreliable. This distinction, which is fundamental in link validation has been considered for the SUDOC scenario as follows. We consider certain the information in the authority records, the information in the bibliographic records but not the link information between bibliographic records and authority records. Therefore, what we aim for is link validation between authority records and bibliographic records. The existing links in the knowledge base (the links we question) induce a partition of bibliographic records according to authority records.

We propose to obtain other partition(s) of such bibliographic notices based on, as much as possible, certain information. This three step process is described in the following. We (1) create contextual authorities that represent the contributors in context of the documents they supposedly wrote. These contextual authorities will be clustered according to different criteria. Once (2) we obtain (one of) the "best" partition(s) (based on combining different criteria clusterings and the according order on the partitions),

we (3) compare its classes with the initial classes given by the authority – bibliographic links. If there is a complete match then the knowledge base links are deemed valid. This decision is heavily relying upon the hypothesis that very similar bibliographic records (cf. different criteria) should be linked to the same authority record. This strong work hypothesis is taken given the (1) workflow librarians use for linking records and (2) the specificity of documentary bases where certain information is reliable (basic author information) and other unreliable (correctness of links towards bibliographic records). Please note that, with this hypothesis, if a person has written very different books (comics and graph theory), we will not be able to detect that both kinds of work pointing to the same person.

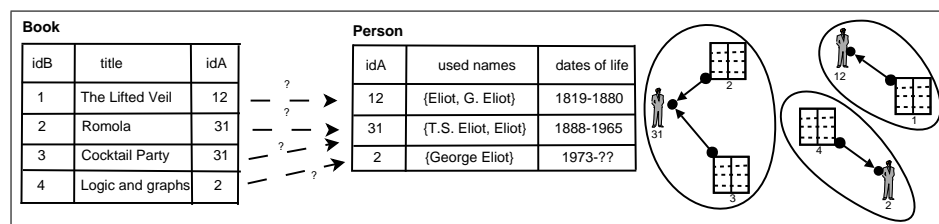


Fig. 1. Two tables with links to be validated

Let us consider the example depicted in Figure 1. The documentary base in question contains two tables: Person and Book. Each of the tables contains some records which might be erroneously linked up. For convenience of intuitive representation, on the right hand side of the picture each of the records in the person table is shown by the means of a man icon and each of books by the means of a book icon. The arrow from a “book” to a “man” aims to represent the uncertain link between the records from two tables. Also, we represent the initial partition classes given by the bibliographic – authority links.

Contextual Contributor-book			
(idA, idB)	title	...	used names
(12,1)	The Lifted Veil		{Eliot, G.Eliot}
(31,2)	Romola		{T.S. Eliot, Eliot}
(31,3)	Cocktail Party		{T.S. Eliot, Eliot}
(2,4)	Logic and graphs		{George Eliot}

Fig. 2. Contextual contributor–book table

Even if links are unreliable, they bring some reliable information. We can assume, that once a bibliographic record was entered in the system, the librarian had the real name of the contributor and thus chose the appropriate authority record willingly. As explained above, the first step is to create a new table containing the joint information from the Person and Book (see Figure 2). This new table will contain records corre-

sponding to the contributor-book information as given by the knowledge base links. We represent this, alternatively, as authors enriched with the book information on the right hand side of the picture.

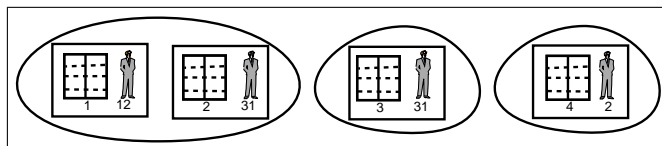


Fig. 3. Partition of contextual contributor-book

The second step would be to cluster these contextual contributor-book records with respect to different criteria (name closeness, domain closeness, publication language etc.) and once (one of the) “best” clustering obtained, to deduce that they correspond to the same real world entity. Such clustering is shown in Figure 3.

We then compare this partition to the initial partition given in Figure 1. As we can see the only link which is validated is the link from book 4 to person 2. For the other links we propose in Section 5 different ways of repair.

3 Link validation in SUDOC base

3.1 Sudoc base

The Sudoc base gathers two kinds of semi-structured records : authority records and bibliographic records. There exists different types of authority records (person, family, corporate body, work etc.). In this paper we focus on person authority records.

A (person) authority record is used to represent a person in the SUDOC knowledge base. In addition to an identifier, it contains at least a set of names used to designate the person and, possibly, dates of birth/death, sex, nationality, titles and any comments in plain text. All the other information regarding its contribution to some works (what he wrote, what domains he has contributed to etc.) are only available from the bibliographic records of the documents (s)he has (co-)authored.

A bibliographic record is used to represent a certain document in the SUDOC knowledge base. The record was created at the moment when the domain expert had a real world document exemplar in front of his eyes. Most the information (such as title, publication date, language, domain) is reliable. The contributor² information is added by searching the system for (person) authority records corresponding to each of the names indicated as contributing of the document. If, for each candidate contributor, the existing documents (contributions) look coherent (for the human eye) with the newly entered record then the reference of the authority record is added in the “contributor” field. The role of the contributor is also added (author, supervisor, editor etc.).

² In this paper we use the term contributor to designate the person who contributed to a document. Author is seen as a specific role of such contributor.

Definition 1 (Authority record). Let $\mathcal{A}_a = \{idP, names, lifeSpan\}$ be the semi-structured schema for a person record. In the following, \mathcal{A}_a^x denotes the domain of the attributes x . \mathcal{A}_a^{idP} corresponds to a Sudoc internal identifier, \mathcal{A}_a^{names} corresponds to lists of strings (the used names for this person), and $\mathcal{A}_a^{lifeSpan}$ correspond to the person's life period. An **authority record** is a tuple $N_a = (idP_{N_a}, names_{N_a}, lifeSpan_{N_a})$ according to this schema. The **set of all authority records** is denoted by \mathcal{N}_a . We also denote by N_a^i the authority record such as $idP_{N_a^i} = i$.

Example 1 (Authority record). Lets describe one of the authority records represented in figure 1. For N_a^{31} :

- $idP_{N_a^{31}} = 31$
- $names_{N_a^{31}} = \{\text{Eliot, T. S. Eliot}\}$
- $lifeSpan_{N_a^{31}} = 1888-1965$

Definition 2 (Bibliographic record). Let $\mathcal{A}_b = \{idB, title, pubDate, lang, domain, contributors\}$ be the semi-structured schema for a bibliographic record. In the following, \mathcal{A}_b^x denotes the domain of the attribute x . \mathcal{A}_b^{idB} corresponds to a Sudoc internal identifier; \mathcal{A}_b^{title} corresponds to a string; $\mathcal{A}_b^{pubDate}$, to a publication date; \mathcal{A}_b^{lang} , to a subset of a finite set of encoded publication languages; \mathcal{A}_b^{domain} , to a subset of a finite set of encoded publication domains. Please note that the contributors attribute corresponds to lists of authority's $idPs$ coupled with their respective roles in the publication ($\mathcal{A}_b^{contributors} = \{(idP, role) | idP \in \mathcal{A}_a^{idP}\}$ and $role \in R$ a finite set of encoded roles). A **bibliographic record** is a tuple $N_b = (idB_{N_b}, title_{N_b}, pubDate_{N_b}, lang_{N_b}, domain_{N_b}, contributors_{N_b})$. The **set of all bibliographic records** is denoted by \mathcal{N}_b . We also denote by N_b^i the bibliography record such as $idB_{N_b^i} = i$.

Example 2 (Bibliographical record). Let describe one of the bibliography records represented in figure 1. For the example we took a recent edition of the book, which explain the late publication date. For N_b^2 :

- $idB_{N_b^2} = 2$
- $title_{N_b^2} = \text{Romola}$
- $pubDate_{N_b^2} = 1997$
- $lang_{N_b^2} = \{\text{English}\}$
- $domain_{N_b^2} = \{\text{"English literature of XIX"}\}$
- $contributors_{N_b^2} = \{(31, author)\}$

$contributors_{N_b} = \{(31, author)\}$ means that the person described by the authority record N_a^{31} (described in example 1) must had contribute to this book as the author.

Definition 3 (Link). Let N_a and N_b an authority, respectively bibliographic record. We say there is a **contributor link** from N_b to N_a typed r (denoted by $N_b \rightarrow_r N_a$) if and only if \exists a role r such that $(idP_{N_a}, r) \in contributors_{N_b}$. We denote $R(N_a, N_b)$ the set of roles which links N_b to N_a . For a given N_a , the set of all bibliographic records linked to N_a , is denoted by $bibliography(N_a)$.

Cf. the above definition, $bibliography(N_a)$ represents all the documents the person represented by N_a has contributed to, according to the SUDOC knowledge base.

Example 3 (Bibliography). According to the figure 1, $\text{bibliography}(N_a^{31}) = \{N_b^2, N_b^3\}$.

We have few information in authority records, but a lot of information in bibliographic records. So, when a librarian would like to link a new bibliographic record N_b^{new} to an existing authority record N_a , (s)he look first at names_{N_a} , and second at the closeness between N_b^{new} and all of the bibliographic records N_b^i as $N_b^i \in \text{bibliography}(N_a)$. Furthermore, links between them are uncertain, but we assume than at least one of the authority record N_a name is the one of the bibliographic record N_b contributor if $N_b \rightarrow_r N_a$. Then, if there is not a single $(\{\text{name of person}\}, N_b^{\text{new}})$ looks like any $(\text{names}_{N_a}, N_b^i) | N_b^i \in \text{bibliography}(N_a)$, N_b^{new} could not be linked to N_a . In order to validate links, we would like to compare the initial partition of links to the best partition of links, according to the closeness of them.

Definition 4. (Initial partition) Let $A_a = \{N_a^1, \dots, N_a^k\} \subseteq \mathcal{N}_a$ a set of k authority records. The **initial partition** of contributor links induced by A_a is defined as: $\mathcal{P}_{\text{initial}}(A_a) = \{P_1, \dots, P_k\}$ such that $\forall i = \overline{1, k}, P_i = \{N_b | N_b \in \text{bibliography}(N_a^i)\}$.

Example 4 (Initial partition). According to the figure 1, $\mathcal{P}_{\text{initial}}(\{N_a^{12}, N_a^{31}, N_a^2\}) = \{\{N_a^{12}\}, \{N_a^{31}, N_a^{31}\}, \{N_a^2\}\}$

We define in section 4 what we would like to partitioning (e.g. contextual authority and information in it), and how to do it.

4 Partitioning method

In the subsection 4.1, we will define the objects to partition for Link Validation problem in the SUDOC case. To decide whether two objects looks like each other, we will use criteria (e.g. subsection 4.3) and, because some criteria are more significant than others, we will use preference relations between them (e.g. subsection 4.5).

4.1 Contextualization of authorities

In order to implement the methodology previously introduced for addressing link validation problem, we need to compute for each document a contextual description of each of its contributors. Such a description, we call a *contextual authority*, will contain a set of selected informations extracted from the bibliographic record and the reliable information about the contributor from the linked authority record. We choose to select from the bibliographic record the title, the publication date, the domain, the language, the co-contributors and the role. Given the plethora of different types of considered attributes, the domain values considered are symbolic (with a total order) in order to accommodate both numerical and discrete values. From the authority record, we consider that only the names as reliable information (up to abbreviations – such as G. for Georges – or involuntary typos) since, at the time of their entry in the system, they were copied from the real world cover of the document.

Definition 5 (Contextual authority). Let N_a an authority record and N_b a bibliographic record such that $N_b \rightarrow_r N_a$. We define the **contextual authority** of N_b according to N_a (denoted $N_b^{(r, N_a)}$): $N_b^{(r, N_a)} = (idB_{N_b}, idP_{N_a}, r, names_{N_a}, title_{N_b}, pubDate_{N_b}, lang_{N_b}, domain_{N_b}, contributors_{N_b})$. We denote by $N_b^{i(r, N_a^j)}$ the contextual authority of N_b^i according to N_a^j .

Example 5 (Contextual authority). Let N_a^{31} an authority record and N_b^2 a bibliographic record respectively described in examples 1 and 2. We have $N_b^2 \rightarrow_{author} N_a^{31}$, then, we can construct the contextual authority $N_b^{2(author, N_a^{31})}$ such as :

- $idB_{N_b^2} = 2$
- $idP_{N_a^{31}} = 31$
- $r = author$
- $names_{N_a^{31}} = \{Eliot, T. S. Eliot\}$
- $title_{N_b^2} = Romola$
- $pubDate_{N_b^2} = 1997$
- $lang_{N_b^2} = \{English\}$
- $domain_{N_b^2} = \{“English literature of XIX^e”\}$
- $contributors_{N_b^2} = \{(31, author)\}$

4.2 Clustering of contextual authorities

All contextual authorities will be then clustered according to different criteria (at least a common name, closeness of domains, dates of publication, languages of publication, others contributors in common or a same title). There is a wide choice of Entity Resolution approaches for solving this problem [6]. The result is a partition of compared objects based on closeness criteria. Please note that given our discrete value approach (as well as a preference order between criteria, explained in 4.3, we can obtain several best partitions.

We choose to implement a clustering method which simulates the librarian decisions when choosing an authority. We only perform the validation on potentially erroneous links rather than on the entire base. The clustering phase is applied on contextual authorities build from a set of authorities sharing at least a common name.

Definition 6 (Partition on contextual authorities). Let $A_a \subseteq \mathcal{N}_a$ a set of authority records. The set of all contextual authorities build from A_a is denoted as:

$$CA(A_a) = \bigcup_{\substack{N_a \in A_a \\ N_b \in \text{bibliography}(N_a) \\ r \in R(N_a, N_b)}} \{N_b^{(r, N_a)}\}$$

The set of all partitions of $CA(A_a)$ is denoted by $\mathbf{B}(CA(A_a))$.

Example 6 (Set of all contextual authorities related to a set au authority records). Let $A_a = \{N_a^2, N_a^{31}\}$ a set of authority records described on figure 1. $CA(A_a) = \{N_b^{4(author, N_a^2)}, N_b^{2(author, N_a^{31})}, N_b^{3(author, N_a^{31})}\}$.

The initial partition given by authority – bibliographic records can be naturally extended to contextual authorities, denoted as $\mathcal{P}_{initial}(CA(A_a))$. This contextual initial partition, chosen wrt criteria described in subsection 4.3, will be used in section 5 for validation.

4.3 Criteria

Criteria used for partitioning are given by librarians. A particularity of our work is the use (cf. librarian needs) of an ordered (\succ_c) discrete value set as $V = \{always, V_{close}, neutral, V_{far}, never\}$ where V_{close} and V_{far} are two totally ordered sets of closeness (respectively farness) values. The bigger (wrt the order \succ_c) an element in V_{close} , the closer the contextual authorities are (in terms of potentially being the same person). The bigger (wrt the order \succ_c) an element in V_{far} the farther the contextual authorities are. The value set above was chosen taking in consideration the following reasons:

- the bipolarity closeness - farness;
- the uncertainty in V_{close} and V_{far} and the certainty of “always” and “never” and
- the lack of information in “neutral”.

For the next examples, we denote :

- $V_{close} = \{\dots, \oplus \oplus \oplus, \oplus \oplus, \oplus\}$, with $\oplus \oplus \oplus \succ_c \oplus \oplus \succ_c \oplus$, and
- $V_{far} = \{\ominus, \ominus \ominus, \ominus \ominus \ominus, \dots\}$, with $\ominus \succ_c \ominus \ominus \succ_c \ominus \ominus \ominus$.

Definition 7 (Criterion). Let $V_C \subseteq V$ and x a given attribute type. A **criterion** C is a function $C : \mathcal{A}^x \times \mathcal{A}^x \rightarrow V_C$. We denote by \mathcal{C} all criteria defined on $\mathcal{A} \times \mathcal{A}$, \mathcal{C}^x all criteria defined on $\mathcal{A}^x \times \mathcal{A}^x$.

Example 7 (A criterion). We consider $CloseByName$ a criterion $\in \mathcal{C}$ as $V_{CloseByName} = \{\oplus \oplus, \oplus, neutral, \ominus\}$. For a, b , two contextual authority names, $CloseByName(a, b) = \oplus \oplus$ means that two a and b are very closed, \oplus that they are close enough and \ominus that they are further. The more a and b are close, the more we prefer to say that the two contextual authority might represent the same person, wrt $CloseByName$ criterion.

Let us consider a criterion on domain closeness (with the assumption that is more likely that the same person wrote books in the same domain). According to the domain values of the contextual authorities, a clustering algorithm will output several partitions of such authorities. Each class of each partition contains contextual authorities with a same / close domain. Let us now consider another criterion on title closeness. This criterion, at its turn, will output a set of partitions on the contextual authorities. How to make sense of all the partitions? Given that the elements inside each class of partition are supposed to refer to the same real world entity, we take the following approach. First, we only consider valid partitions (partitions that do not have either inside a class “impossible” closeness values or between two elements of two different classes “always” closeness values). Then, for each criterion, the partition will have as its value a tuple containing the (1) value – wrt element comparison – inside partition classes (how well it gets elements together) and (2) value – wrt element comparison – between two different classes (how well it splits far elements). Based on the values in this tuple we

define a partial order between partitions. The partition partial order, on one criterion, is then extended to two (or more criteria).

For the SUDOC example, we could use several criteria as *CloseByName*, *Role*, and *CommonContributors*. However, for the rest of this question, we will use an abstract example with two criteria a and b (represented figure 4) for develop examples on partition choice wrt several criteria.

4.4 Partition wrt a criterion

We consider partitions value wrt a criterion only for valid partitions wrt a set of criteria C_a , because, if a partition is not valid, it is not an acceptable solution.

Definition 8 (Valid partition). Let $\mathcal{P} = \{P_1, \dots, P_n\} \in \mathbf{B}(CA(A_a))$ a partition of contextual authorities build from A_a . \mathcal{P} is said a **valid partition** for the criteria set $C_a \subseteq \mathcal{C}$ if and only if:

- $\forall x_1, x_2 \in CA(A_a)$, with $x_1 \in P_i, x_2 \in P_j, i \neq j$ and $\forall x \in \mathcal{A}, \nexists C \in (\mathcal{C}^x \cap C_a)$ such that $C(x[x_1], x[x_2]) = \text{always}$, where $x[x_i]$ represents the value of the attribute of type x for the contextual authority x_i .
- $\forall x_1, x_2 \in CA(A_a)$, with $x_1, x_2 \in P_i$, and $\forall x \in \mathcal{A}, \nexists C \in (\mathcal{C}^x \cap C_a)$ such that $C(x[x_1], x[x_2]) = \text{never}$.

Example 8 (Valid partition). Let a, b be two criterium, and A, B, C, D, E be contextual authorities. We present the comparison values of criteria a and b on the graph 1 respectively 2 on figure 4. In this example, any partition of $\{A, B, C, D, E\}$ is valid for $C_a = \{a, b\}$ because there is no 'never' or 'always' values of comparison for a and b criteria.

Please note that if two contextual authorities that should be grouped together (because they might point to the same entity) have a big value returned by a criterion, we are more in agreement with the class of partition containing them both than otherwise. This "agreement" notion will be used in the following definition where the value of the partition is introduced. If elements in the same class have a bigger closeness value over V the more we agree. If elements in two different classes have a smaller farness value over V the more we agree. Therefore, we define \succ_a , an agreement partial order on V such that $\text{always} \succ_a V_{\text{close}} \succ_a \text{neutral}$ and $\text{never} \succ_a V_{\text{far}} \succ_a \text{neutral}$ (the order on V_{close} is the same as \succ_c and on V_{far} is the reverse order as \succ_c).

Definition 9 (Partition value wrt to a criterion). Let $\mathcal{P} = \{P_1, \dots, P_n\} \in \mathbf{B}(CA(A_a))$ a partition of contextual authorities build from A_a and $C \in \mathcal{C}^x$ a criterion. The **value** $Val_C(\mathcal{P})$ of the partition \mathcal{P} wrt to criterion C , is the tuple (intra, inter) where:

- $\text{intra} = \max_{\succ_a} \{V_{\text{close}} \cup \{\text{neutral}\} \cap \{C(x[x_1], x[x_2])\}\},$
 $\forall i, j \in \{1, \dots, n\}, i \neq j, \forall x_1 \in P_i \text{ and } x_2 \in P_j.$
- $\text{inter} = \max_{\succ_a} \{V_{\text{far}} \cup \{\text{neutral}\} \cap \{C(x[x_1], x[x_2])\}\},$
 $\forall i \in \{1, \dots, n\} \forall x_1, x_2 \in P_i.$

Example 9 (Partition value wrt to a criterion). For the example figure 4, let's have four partitions $\mathcal{P}_{\text{all}}, \mathcal{P}_1, \mathcal{P}_2, \mathcal{P}_{\text{none}}$.

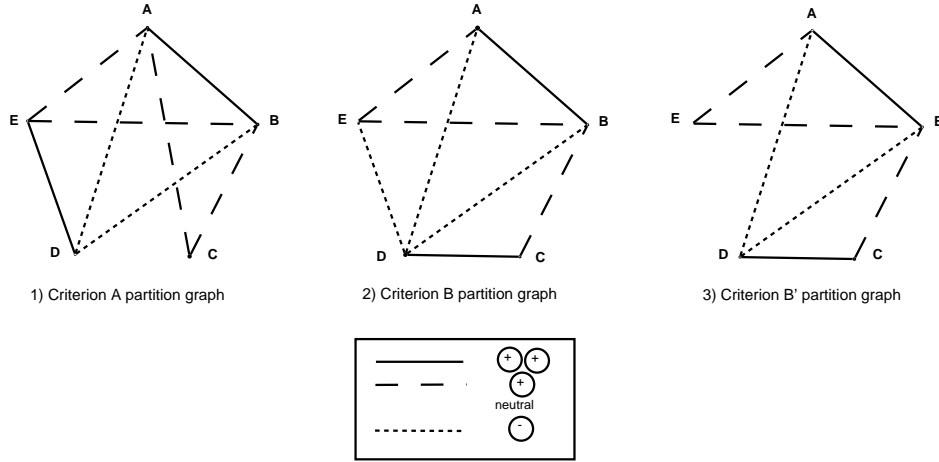


Fig. 4. Criteria graphs

- $\mathcal{P}_{all} = \{\{A, B, C, D, E\}\} : Val_a(\mathcal{P}_{all}) = Val_b(\mathcal{P}_{all}) = (neutral, \ominus)$
- $\mathcal{P}_1 = \{\{A, B, C\}, \{D, E\}\} : Val_a(\mathcal{P}_1) = (\oplus, neutral) \text{ and } Val_b(\mathcal{P}_1) = (\oplus\oplus, \ominus)$
- $\mathcal{P}_2 = \{\{A, B, E\}, \{C, D\}\} : Val_a(\mathcal{P}_2) = (\oplus\oplus, neutral) \text{ and } Val_b(\mathcal{P}_2) = (\oplus, neutral)$
- $\mathcal{P}_{none} = \{\{A, B, C, D, E\}\} : Val_a(\mathcal{P}_{none}) = Val_b(\mathcal{P}_{none}) = (\oplus\oplus, neutral)$

We introduce a partial order between two partitions with respect to one criterion. This order will privilege the the fact that we separate contextual authorities who have “reasons” to be apart:

Definition 10 (Partition order wrt to a criterion). Let $\mathcal{P}_1, \mathcal{P}_2 \in \mathbf{B}(CA(A_a))$ two valid partitions of contextual authorities build from A_a and $C \in \mathcal{C}^x$ a criterion. Given $Val_C(\mathcal{P}_1) = (intra_1, inter_1)$ and $Val_C(\mathcal{P}_2) = (intra_2, inter_2)$, we define $better_C^{nr} \subseteq \mathbf{B}(CA(A_a)) \times \mathbf{B}(CA(A_a))$ such that $\mathcal{P}_1 better_C^{nr} \mathcal{P}_2$ if and only if:

- $inter_1 \succ_a inter_2$ or
- if $inter_1$ and $inter_2$ have the same value then $intra_1 \succ_a intra_2$.

Example 10 (Partition order wrt to a criterion). Let a set of contextual notices $CA(A_a) = \{A, B, C, D, E\}$, and $a, b \in \mathcal{C}$, two criteria. The comparisons values on $\{A, B, C, D, E\}$ wrt a and b are shown on graph 1 and respectively graph 2 on figure 4. Let $\mathcal{P}_1, \mathcal{P}_2$ partitions $\in \mathbf{B}(CA(A_a))$ described on example 9. We have $\mathcal{P}_1 better_a^{nr} \mathcal{P}_2$ and $\mathcal{P}_2 better_b^{nr} \mathcal{P}_1$.

We can choose a better partition wrt a criterion $C \in \mathcal{C}$. According to the order $better_C$ we denote the chosen best partition with respect to criterion C by $Best_{\{C\}}^{A_a}$.

But librarians using several criteria to decide whether a bibliographical record must be linked to the same authority record than an other one (as name, others common contributors, language, domains...). Furthermore, a criterion could be more significant than an other (as name is more significant than language), or complementary with an

other one (as common contributors and domain ³). That's why we need several criteria and preferences relations between criteria to modeling how a librarian decide wether two bibliographic records are close.

Let's define relation between them and how to choose a better partition wrt to two criteria.

4.5 Preferences between criteria

In order to modeling how a librarian decide wether two bibliographic records must belong to the same person because they are very close (according to several criteria), we need to say how we consider a criterion compared with an other criterion.

Definition 11 (Complementary, Priority, Paribus Ceteris Priority).

On $\mathcal{C} \times \mathcal{C}$ we define the following relations:

- $\diamond_c \subseteq \mathcal{C} \times \mathcal{C}$, *reflexive, symmetrical and transitive*
- $>_c \subseteq \mathcal{C} \times \mathcal{C}$, *anti-symmetrical and transitive*
- $\gg_c \subseteq \mathcal{C} \times \mathcal{C}$, *anti-symmetrical and transitive*

The first relation is the complementary relation (\diamond_c). Intuitively this relation will be used for representing the "help" one criterium can give to the other for better entity distinguishing. It is enough for ER to use one criterium or the other but having them both will result in a better ER process. It is enough we have satisfied either the same domain criterium or the co-author criterium. Either (or both) of criteria is enough for deduplication.

The second is the priority relation ($>_c$). For the same values we consider first the priority criterium. SameName and domain. Ideally the authors should have the same name AND the same domain but the name is a better criteria than the domain.

The third is the Paribus Ceteris priority relation (\gg_c). If same values for C_i then we look for C_j . Language and contributors. The contributors is a good criterion to deduplicate. If we have doubts (several partitions with equal values wrt contributor criterion) we can use the language criterion. But the language criterion is useless if the co-author criteria prefer one partition on the other ones.

Definition 12. *Let $A, B, C \in \mathcal{C}$. The relations $\diamond_c, >_c, \gg_c$ satisfy the following :*

- $\diamond_c \cap >_c = \emptyset; \diamond_c \cap \gg_c = \emptyset$
- $A >_c B$ et $B \gg_c C$, then $A \gg_c C$;
- $A \gg_c B$ et $B >_c C$, then $A \gg_c C$;
- $A >_c B$ et $A \diamond_c C$, then $C >_c B$;
- $A >_c B$ et $B \diamond_c C$, then $A >_c C$;
- $A \gg_c B$ et $A \diamond_c C$, then $C \gg_c B$;
- $A \gg_c B$ et $B \diamond_c C$, then $A \gg_c C$.

³ Common contributors and domain criteria give information on publication context. The satisfaction of one of these criteria is enough to consider than the publication context is close between two contextual authorities.

Let a set of criteria $\mathcal{C}_a \subseteq \mathcal{C}$ and a partition $\mathcal{P} \in \mathbf{B}(CA(A_a))$. If \mathcal{P} is not valid (e.g. definition 8) for $C_j \in \mathcal{C}_a$, \mathcal{P} is not valid for \mathcal{C}_a , and we are not interested by its value wrt several criteria because \mathcal{P} cannot be a solution. In the following subsection, we will compare the value of several valid partitions wrt to two criteria.

4.6 Partition Evaluation wrt Preference Enriched Criteria

Please notice that the partition order wrt to a criterion, better^{nr} relation is not reflexive according to the definition 10. However, for the following we will need better to be reflexive. We define then $\forall i, \text{better} = \text{better}^{nr} \cup (\mathcal{P}_i, \mathcal{P}_i)$.

Definition 13 (The best partition wrt to two Independent Criteria).

Let $C_1, C_2 \in \mathcal{C}$ two **independent criteria** and $\mathcal{P}_1, \mathcal{P}_2$, two partitions such as $\mathcal{P}_1, \mathcal{P}_2 \in \mathbf{B}(CA(A_a))$. $\mathcal{P}_1 \text{better}_{C_1, C_2} \mathcal{P}_2$ if and only if:

- $\mathcal{P}_1 \text{better}_{C_1} \mathcal{P}_2$ and $\mathcal{P}_1 \text{better}_{C_2} \mathcal{P}_2$, or
- $\text{Val}_{C_i}(\mathcal{P}_1) = (\text{inter}, \text{"neutral"}) \forall i=1,2$ and $\exists i=1,2$ such that $\text{Val}_{C_i}(\mathcal{P}_2) = (\text{inter}, \text{intra})$ where $\text{intra} \in V_{far}$.

Example 11 (Criteria independency effect on partitioning). Let a set of contextual notices $CA(A_a) = \{A, B, C, D, E\}$, and $a, b \in \mathcal{C}$, two criteria. The comparisons values on $\{A, B, C, D, E\}$ wrt a and b are shown on graph 1 and respectively graph 2 on figure 4. Let $\mathcal{P}_1, \mathcal{P}_2$ partitions $\in \mathbf{B}(CA(A_a))$ described on example 9. We consider a and b independents of each other. Then, $\mathcal{P}_2 \text{better}_{a,b} \mathcal{P}_1$.

Definition 14 (The best partition wrt to two complementary Criteria).

Let $C_1, C_2 \in \mathcal{C}$ two **complementary criteria** and $\mathcal{P}_1, \mathcal{P}_2$ defined as above two partitions in $\mathbf{P}(\text{bibliography}_{1,\dots,i})$. We compute \mathcal{P}_i^\diamond , $i = 1, 2$ as follows: $\forall o_1, o_2 \in \text{bibliography}_{1,\dots,i}$ if $C_1(o_1, o_2) \in V_{far}$ ⁴ and $C_2(o_1, o_2) \in V_{close}$ then $C_1(o_1, o_2) = \text{"neutral"}$.

Then, $\mathcal{P}_1 \text{better}_{C_1, C_2} \mathcal{P}_2$ if and only if $\mathcal{P}_1^\diamond \text{better}_{C_1, C_2} \mathcal{P}_2^\diamond$.

Example 12 (Criteria complementarity effect on partitioning). Let a set of contextual notices $CA(A_a) = \{A, B, C, D, E\}$, and $a, b \in \mathcal{C}$, two criteria. The comparisons values on $\{A, B, C, D, E\}$ wrt a and b are shown on graph 1 and respectively graph 2 on figure 4. We consider that $a \diamond_c b$. The graph 3 on figure 4 shows the comparison values of $\{A, B, C, D, E\}$ we use to obtain the value of a partition \mathcal{P} wrt the complementary relations. Let $\mathcal{P}_1 = \{\{A, B, C\}, \{D, E\}\}$, a partition $\in \mathbf{B}(CA(A_a))$ described on example 9.

- $\text{Val}_b(\mathcal{P}_1) = (\oplus \oplus, \ominus)$ (without complementary relations)
- $\text{Val}_b(\mathcal{P}_1) = \text{Val}_b(\mathcal{P}_1^\diamond) = (\oplus \oplus, \text{neutral})$ (with $a \diamond_c b$)

Definition 15 (The best partition wrt to two priority Criteria).

Let $C_1, C_2 \in \mathcal{C}$ two **priority criteria** such that $C_1 > C_2$ and $\mathcal{P}_1, \mathcal{P}_2$ defined as above two partitions. $\mathcal{P}_1 \text{better}_{C_1, C_2} \mathcal{P}_2$ if and only if:

⁴ The notation $C_1(o_1, o_2) \in V_{far}$ corresponds to $\exists A_1, A_2 \in \mathcal{A}^x$, two values of the common attribute of type bibliographical notices o_1 and o_2 , where $C_1(A_1, A_2) \in V_{far}$

- $Val_{C_i}(\mathcal{P}_1) = (\text{inter}, \text{"neutral"}) \forall i=1,2$ and $\exists i=1,2$ such that $Val_{C_i}(\mathcal{P}_2) = (\text{inter}, \text{intra})$ where $\text{intra} \in V_{far}$.
- $\mathcal{P}_1 \text{better}_{C_1}^{nr} \mathcal{P}_2$ or, if equal, $\mathcal{P}_1 \text{better}_{C_2} \mathcal{P}_2$.

Example 13 (Priority between criteria effect on partitioning). Let a set of contextual notices $CA(A_a) = \{A, B, C, D, E\}$, and $a, b \in \mathcal{C}$, two criteria. The comparisons values on $\{A, B, C, D, E\}$ wrt a and b are shown on graph 1 and respectively graph 2 on figure 4. Let $\mathcal{P}_1, \mathcal{P}_2$ partitions $\in \mathbf{B}(CA(A_a))$ described on example 9. We consider $a > b$. Then, $\mathcal{P}_2 \text{better}_{a,b} \mathcal{P}_1$.

Definition 16 (The best partition wrt to two Paribus Ceteris priority Criteria).

Let $C_1, C_2 \in \mathcal{C}$ two **Paribus Ceteris priority criteria** such that $C_1 > C_2$ and $\mathcal{P}_1, \mathcal{P}_2$ defined as above two partitions. $\mathcal{P}_1 \text{better}_{C_1, C_2} \mathcal{P}_2$ if and only if:

- $\mathcal{P}_1 \text{better}_{C_1}^{nr} \mathcal{P}_2$ or, if equal, $\mathcal{P}_1 \text{better}_{C_2} \mathcal{P}_2$.

Example 14 (Paribus ceteris priority between criteria effect on partitioning). Let a set of contextual notices $CA(A_a) = \{A, B, C, D, E\}$, and $a, b \in \mathcal{C}$, two criteria. The comparisons values on $\{A, B, C, D, E\}$ wrt a and b are shown on graph 1 and respectively graph 2 on figure 4. Let $\mathcal{P}_1, \mathcal{P}_2$ partitions $\in \mathbf{B}(CA(A_a))$ described on example 9. We consider $a \gg b$. Then, $\mathcal{P}_1 \text{better}_{a,b} \mathcal{P}_2$.

Please note that since (1) the discrete values are used for criterion values and (2) the different criteria used for comparison are in a partial preference order, we can obtain several best partitions. In the next section, the chosen overall best partition for a set of authorities records A_a and a set of criteria \mathcal{C}_a is denoted by $\text{Best}_{\mathcal{C}_a}^{A_a} \in \mathbf{B}(CA(A_a))$. $\text{Best}_{\mathcal{C}_a}^{A_a}$ is valid for \mathcal{C}_a .

5 Link Validation Repair

If the classes of partitions correspond to the initial partition corresponding to the contributor links then these links are deemed valid (it means all the contributors picked by the librarians are exactly the ones chosen by the clustering algorithm). If not, repairing options for certain cases are proposed.

For this section, we will use $\text{Best}(A_a, \mathcal{C}_a)$, the best partition of bibliography records deduced from $\text{Best}_{\mathcal{C}_a}^{A_a}$ such as :

- $\text{Best}_{\mathcal{C}_a}^{A_a} = \{P_1, \dots, P_k\}$
- $\text{Best}(A_a, \mathcal{C}_a) = \{P'_1, \dots, P'_k\}$
- $N_b^i, N_b^j \in P'_m \in \text{Best}(A_a, \mathcal{C}_a)$ if and only if $\exists N_b^{i(r, N_a)}, N_b^{j(r', N'_a)} \in P_m \in \text{Best}_{\mathcal{C}_a}^{A_a}$

Definition 17 (Link valid). Let $A_a \subseteq \mathcal{N}_a$ be a set of authority records. A class $P \in \mathcal{P}_{\text{initial}}(A_a)$ is **link valid** for a set of criteria $\mathcal{C}_a \subseteq \mathcal{C}$ if and only if $\exists P' \in \text{Best}(A_a, \mathcal{C}_a)$ such that $P' = P$.

If a class of partition is link valid, then, consequently, all the links underlying this class are deemed valid. Furthermore, if all the classes of the initial partition are link valid then the knowledge base is deemed link valid.

Example 15 (Class Linked valid). Let $A_a = \{N_a^{12}, N_a^{31}, N_a^2\}$ be a set of authority records, and $C_a \subseteq C$, be a set of criteria. From figures 1 and 3, we have :

- $\mathcal{P}_{initial}(A_a) = \{\{N_b^1\}, \{N_b^2, N_b^3\}, \{N_b^4\}\}$ (e.g. example 4), and
- $\mathcal{B}_{est}(A_a, C_a) = \{\{N_b^1, N_b^2\}, \{N_b^3\}, \{N_b^4\}\}$ (e.g. example 3).

$P = \{N_b^4\}$ is a class as $P \in \mathcal{P}_{initial}(A_a)$ and $P \in \mathcal{B}_{est}(A_a, C_a)$. So, P is linked valid, and confirm that N_b^4 points to N_a^2 .

If the knowledge base is not link valid, we give three straightforward repair possibilities. More repair scenarios are investigated and are an inherent part of current / future work. All of the below reparation heuristics are based on the following two librarian inspired assumptions:

- we only redirect bibliography –authority links within the same class of partition or to a newly created authority and
- we cannot redirect all bibliographic records from an authority record.

The first repair heuristic is fusion. We do this when a class P is linked to two authority records, where this two authority record are only linked to P . We fusion authority records in a joint authority record and the links will point to the newly created authority record. Some authority records will thus disappear. $fusion(N_a^i, N_a^j) = N_a^{new}$ and $bibliography(N_a^{new}) = bibliography(N_a^i) \cup bibliography(N_a^j)$.

Example 16 (Fusion). Let $A_a = \{N_a^{12}, N_a^{31}, N_a^2\}$ be a set of authority records, and $C_a \subseteq C$, be a set of criteria. We consider :

- $\mathcal{P}_{initial}(A_a) = \{\{N_b^1\}, \{N_b^2, N_b^3\}, \{N_b^4\}\}$ (e.g. example 4), and
- $\mathcal{B}_{est}(A_a, C_a) = \{\{N_b^1, N_b^2, N_b^3\}, \{N_b^4\}\}$.

$P = \{N_b^1, N_b^2, N_b^3\}$ is a classe $\in \mathcal{B}_{est}(A_a, C_a)$ and P is linked to two authority records N_a^{12} and N_a^{31} (e.g. example 4). According to our best partition, we link N_b^1, N_b^2 and N_b^3 to $fusion(N_a^{12}, N_a^{31})$ then erase N_a^{12} and N_a^{31} .

The second repair is when two classes P_1 and P_2 of our best partition point to the same authority record N_a . In this case we create two new authority records that replace the initial one and redirect links to the two authority records (in order to characterize this newly created authority records certain techniques such as the Generation of Referring Expressions can be used [8]): N_a^{new1} and N_a^{new2} with $bibliography(N_a^{new1}) = \{N_b | N_b^{(r, N_a)} \in P_1\}$ and $bibliography(N_a^{new2}) = \{N_b | N_b^{(r, N_a)} \in P_2\}$.

Example 17 (Deduplication of a authority record). Let $A_a = \{N_a^{12}, N_a^{31}, N_a^2\}$ be a set of authority records, and $C_a \subseteq C$, be a set of criteria. We consider :

- $\mathcal{P}_{initial}(A_a) = \{\{N_b^1\}, \{N_b^2, N_b^3\}, \{N_b^4\}\}$ (e.g. example 4), and
- $\mathcal{B}_{est}(A_a, C_a) = \{\{N_b^1\}, \{N_b^2\}, \{N_b^3\}, \{N_b^4\}\}$.

$P_1 = \{N_b^2\}, P_2 = \{N_b^3\}$ are classes $\in \mathcal{Best}(A_a, \mathcal{C}_a)$ and P_1, P_2 are linked to the same authority record N_a^{31} (e.g. example 4). According to our best partition, we create two new authority records, N_a^{31new1} and N_a^{31new2} as $\text{bibliography}(N_a^{31new1}) = P_1$ and $\text{bibliography}(N_a^{31new2}) = P_2$. We erase N_a^{31} .

Last, let P , a class of the best partition and N_a , an authority record such as $\text{bibliography}(N_a) \subseteq P$. We also have a bibliographical record N_b pointing to an other authority record N'_a such as $\text{bibliography}(N'_a) \not\subseteq P$ and $N_b \in P$. Then we redirect N_b towards N'_a :

- $\text{bibliography}(N_a) = \text{bibliography}(N_a) \cup \{N_b | N_b \in (P \cap \text{bibliography}(N'_a))\}$,
- and $\text{bibliography}(N'_a) = \text{bibliography}(N'_a) \setminus \{N_b | N_b \in P\}$

Example 18 (Link redirection). Let $A_a = \{N_a^{12}, N_a^{31}, N_a^2\}$ be a set of authority records. We consider :

- $\mathcal{P}_{\text{initial}}(A_a) = \{\{N_b^1\}, \{N_b^2, N_b^3\}, \{N_b^4\}\}$ (e.g. example 4), and
- $\mathcal{Best}(A_a, \mathcal{C}_a) = \{\{N_b^1, N_b^2\}, \{N_b^3\}, \{N_b^4\}\}$ (e.g. example 3).

$P_1 = \{N_b^1, N_b^2\}$ is a classe $\in \mathcal{Best}(A_a, \mathcal{C}_a)$. $P_1 = \text{bibliography}(N_a^{12}) \cup \{N_b^2\}$ and $N_b^2 \in \text{bibliography}(N_a^{31})$. So, we link N_b^2 to N_a^{12} instead of N_a^{31} .

Please note that the techniques described above are the simple intuitive cases for repair. Further scenarios can be envisaged and are currently pursued by our work.

6 Conclusion

In this paper we investigated the link validation problem, inspired by a joint project with ABES. We pointed out the fundamental difference with the Entity Resolution problem and proposed a partitioning based approach for addressing it.

There are a number of interesting features of presented work. First, the criteria used for partitioning have the value set a totally ordered discrete set. Therefore certain numerical operations (addition, etc.) are not straightforward applicable (while very common in clustering). Thus, our “real world relatedness” intuitive semantics should be transformed into certain syntactic operations on these partitions. Certain numerical measures currently used in clustering (such as Rand’s measure[10]) where intra and inter class values are considered should be adapted to our semantic needs. Also, approaches such as Dedupalog [1] consider clustering symbolical values. However, they do not consider different levels of closeness and farness (only considering one positive value and one negative value). Second, the (multi criteria) partitioning problem can easily be seen as a (colored) clique problem on the graph representing enriched bibliographic records as nodes. This could pave the way to interesting graph theoretic optimisations of finding the best partitions. Third, the different preference relation amongst criteria are to be investigated for further optimisation.

Last and not least, this approach needs implemented and tested. While, on the implementation end optimisation is essential (given the large number of possible partitions a naive approach takes into account), the evaluation should be thoroughly thought out in close contact with librarians.

7 Acknowledgements

We acknowledge the support of ABES for this work.

References

1. A. Arasu, C. Ré, and D. Suciu. Large-scale deduplication with constraints using dedupalog. In *in: Proceedings of the 25th International Conference on Data Engineering (ICDE)*, pages 952–963, 2009.
2. O. Benjelloun, H. Garcia-Molina, D. Menestrina, Q. Su, S. Whang, and J. Widom. Swoosh: a generic approach to entity resolution. *The VLDB Journal*, 18:255–276, 2009.
3. I. Bhattacharya and L. Getoor. *Entity Resolution in Graphs*, pages 311–344. John Wiley & Sons, Inc., 2006.
4. P. Bouquet, H. Stoermer, and B. Bazzanella. An entity name system (ens) for the semantic web. In *Proceedings of the 5th European semantic web conference on The semantic web: research and applications, ESWC’08*, pages 258–272, Berlin, Heidelberg, 2008. Springer-Verlag.
5. S. Chaudhuri, K. Ganjam, V. Ganti, and R. Motwani. Robust and efficient fuzzy match for online data cleaning. In *Proceedings of the 2003 ACM SIGMOD international conference on Management of data, SIGMOD ’03*, pages 313–324, New York, NY, USA, 2003. ACM.
6. A. K. Elmagarmid, P. G. Ipeirotis, and V. S. Verykios. Duplicate record detection: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 19:1–16, 2007.
7. M.-Y. Kan and Y. F. Tan. Record matching in digital library metadata. *Commun. ACM*, 51:91–94, February 2008.
8. N. Moreau, M. Leclère, and M. Croitoru. Distinguishing answers in conceptual graph knowledge bases. In *Proc. of ICCS: Conceptual Structures: Leveraging Semantic Technologies*, pages 233–246. LNAI Springer, 2009.
9. H. B. Newcombe, J. M. Kennedy, S. J. Axford, and A. P. JAMES. Automatic linkage of vital records. *Science*, 130:954–959, Oct. 1959.
10. W. Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical association*, pages 846–850, 1971.
11. F. Saïs, N. Pernelle, and M.-C. Rousset. Reconciliation de references : une approche logique adaptee aux grands volumes de donnees. In *EGC*, pages 623–634, 2007.
12. N. R. Smalheiser and V. I. Torvik. *Annual Review of Information Science and Technology (ARIST)*, volume 43, chapter Author Name Disambiguation. Information Today, Inc, 2009.
13. W. E. Winkler. Overview of record linkage and current research directions. Technical report, BUREAU OF THE CENSUS, 2006.
14. S. Wölger, C. Hofer, K. Siorpaes, S. Thaler, E. Simperl, and T. Bürger. Interlinking data - approaches and tools. Technical report, STI Innsbruck, University of Innsbruck, 2011.