SudocAD: A Knowledge-Based System for Object Identification

Michel Chein, Michel Leclère, Yann Nicolas

SudocAD: A Knowledge-Based System for Object Identification

Michel Chein and Michel Leclère and Yann Nicolas

RR LIRMM-GraphIK December 14, 2012

Abstract

This is a DRAFT version of a paper that will be finalized and submitted for publication in January 2013

1 Introduction

The problem addressed in this paper is an object identification problem. Since a long time, ([**?**] is a seminal paper published in 1959) various problems having at their core identification problems have been extensively studied under different names (cf. the bibliography [**?**]) such as:

- *entity resolution, reference reconciliation*: identifying multiple references to the same object and distinguishing them from mentions of different objects, finding which records in two different databases represent the same entity
- *de-duplication*: finding duplicate data (object) then deleting the duplicates (Classical problem: creation of mailing lists)
- *merging* merging records judged to represent the same world entity
- *record linkage*, linking records through references to same world entities
- *entity alignment* this term is issued from analogy with ontology alignment and is used in the context of web of objects (?)
- *object identification* is used in this paper as a generic term to single out, to distinguish, to recognize in a computer system symbols aiming at representing the same individual entity in the world modeled in the system

These problems have been considered for many kinds of databases, especially data warehouses where, for instance, it is important to know if two identifiers refer to the same object or to different entities in the exterior world. These problems acquired a new importance due to the web, especially in the so-called web of objects, where, for instance, one wants to gather all available information concerning the same entity.

Most solutions of object identification problems are based on classification techniques. In such an approach, an entity is described by a list of attributes; attribute values are simple data types (e.g., strings or numbers) and approximate similarity measures are assigned for each kind of attributes vector; a similarity measure is built for lists of attributes (often it is a weighted combination of the attribute similarities); finally, a decision procedure allows to state when two lists of attributes represent the same entity. Recently, logical approaches using knowledge representation techniques (e.g., Fatiha Saïs, Nathalie Pernelle and M.-C. Rousset consider data sources conforming to the same RDFS schema [**?**, **?**] ...) and combinations of these two kinds of methods have been developed (...).

Our method belongs to this latter type, it combines numerical measures for comparing low level attributes and for building symbolic relations and uses logical rules for processing these qualitative relations. The languages of the semantic web, RDFS and OWL, and knowledge representation techniques have been used for developing the system aiming at solving a specific object identification in a context of bibliographic databases.

The problem and the method described in this article can be sketchily stated as follows.

Having a database containing information about entities $\mathcal{E} = \{E_1, \ldots, E_n\}$ of a specific type T, e.g. $Person$, and having a (new) piece of information d concerning an entity E of type T, i.e. a $Person$, one wants to find if there is $E_i \in \mathcal{E}$ such that E and E_i refer to the same person. Our approach consists in computing a qualitative partition of $\mathcal{E}$, whose classes called $\{StrongLinkage, MediumLinkage, \ldots, ImpossibleLinka$ are ordered by decreasing relevance with respect to co-reference with E. This result can be used as follows. If there is only one element E' in the class $StrongLinkage$ then the system can automatically conclude that $E = E'$, otherwise the classes can be successively presented to a human operator who will have to choose. In the bibliographic context considered, the database contains (at least) authority databases (one for each type of authority, e.g. Person, Collective entity, Geographical place, ...) containing data about the authorities, and a document database containing metadata about documents and in particular references to the authorities. Thus, $\mathcal{E}$ is composed of a base $\mathcal{A}$ of authority notices gathering general information concerning a specific type of entities and a base $\mathcal{D}$ containing document notices referring to entities in $\mathcal{E}$. If the initial database is not structured in such a way, the method proposed can be nevertheless applied if it is possible to build $\mathcal{E}$ and $\mathcal{D}$ from it (cf. section 3.2 that can be adapted to do that).

Let us mention some reasons enlightening the importance of object identification problems in document databases. Firstly, all the previous mentioned problems are important in the context of libraries: some of them are induced by the evolution of libraries (e.g., adding notices to a base, merging bibliographic bases, maintenance of the different bases), others concern the quality of the notice bases (e.g., consistency inside and between bases, relevance of the subject). Secondly, a document database is a very rich source of information about the documents themselves and about authorities. International work is done to standardize the metadata (FRBR, CIDOC CRM, RDA, ?), to build shared ontologies. This allows the transformation of a document base into a knowledge base and then knowledge-based techniques can be used to make the most of the information present in the base. Thirdly, whenever the languages of the semantic web are used for solving object identification problems (as it is the case in the system described in this paper) and due to the large size of document bases, the techniques developed are not only interesting *per se* but also as a testbed for object identification in the context of the web of data.

The paper is organized as follows. In section 2, the bibliographic database context and the goals of the system sudocAD are precisely stated. In particular, how a bibliographic database has been represented as a knowledge base by using the web standards. The hybrid method, combining numerical and logical aspects, is presented in section 3. The results of the evaluation of sudocAD are described in section 5. Specificities of the system sudocAD are presented section 4. Conclusion and further work end the paper. Note that in this paper the term sudocAD is used for the system described section 3 and section 4. The version presented is the one that has been evaluated, the methodology of the evaluation and the results are presented in section 4. Many improvements have been considered during the development of sudocAD and some of them are presented in the last section. We are presently investigating what are the changes and improvements mandatory to make it usable in production ...

2 The Bibliographic Context

2.1 Bibliographic and Authority Notices, sudoc and IdRef

Sudoc is the national bibliographic infrastructure for the French higher education. As such an infrastructure, Sudoc is both:

- a shared database where bibliographic records are created once, but used by all ;
- a network of professional catalogers, with common tools, guidelines and tools.

The core function of this collective endeavour is to create and maintain database records that describe the documents held or licensed by French Universities and other higher education institutions. Sudoc contains more than ten million such records (2012 figure). Documents described by Sudoc are mainly electronic or print books and journals, but also manuscripts, pictures, maps, etc.

A Sudoc record is composed of three kind of information:

- Meta-information
- Descriptive information
- Access points

Meta-information is information about the record. It is out of the scope of this paper. Descriptive information is mere transcription of information that is found in the document to be catalogued. By instance, the descriptive field Title contains the very text string that is written on the title page. The same is true for the descriptive field Author: the record has to keep the author's name as it is found in the document, even if the cataloger knows that the title page misspelled this name. This strictly descriptive approach aims to identify without any ambiguity the publication. The transcribed information is to be sufficient in order to distinguish two editions of the same work. But this descriptive approach may make the document harder to find. If the title page, hence the bibliographic record, assumes that the author's name is "Jean Dupond" whereas the actual name is "Jean Dupont", the library users that are only aware of the actual name will fail to find and access the record in the catalog and then to find and access the document in the library. In order to prevent this kind of problem, the cataloguing rules prescribe to add the actual name in the record, not instead of but besides the one found on the title page. This kind of additional non descriptive information is called "access point".

Access points constitute the third kind of information that is to be found in a bibliographic record. An access point is a piece of information that focuses on some aspect of the described document that may be relevant for finding the record in a database and that is not directly observable in the document itself. As we have seen above, it can be the actual author's name. It can also be the author's name written according to a specific convention (e.g. "Dupont, Jean" or "Dupont, Jean (1956-)"). But an access point is not necessarily an alternative textual form of a piece of information that was observed on the described object. It can be the result of the analysis of the content of the document. By instance, an access point can be a keyword expressing the subject of the document (e.g. "cars"). Any bibliographic record is the result of the description of the document and the selection of relevant access points. But when is a bibliographic deemed to be achieved? Theoretically, the description could last forever but cataloguing guidelines fix a finite list of which document properties must or may be described. But regarding access points, when to stop? If access points are there to help the user to find the record, there can be plenty of them. Should the record describing the book by Jean Dupont and about cars have access points mentioning all the variant of Jean Dupont's name and as many access points as there are synonyms for "cars"? And should it be the case for each book by Jean Dupont or about cars? In order not to repeat all the variants of a name or a concept in all the bibliographic records using this name or this concept as an access point, theses variants are grouped in specific records, namely authority records. By instance, the authority record for Jean Dupont will contains his name variants. One of these variants will be distinguished as the preferred form. Some guidelines will help the cataloguers to choose the preferred form. Some guidelines expect the preferred form to be unique. If true, the preferred form will be used as if it was an identifier. The preferred form is the only form to be used an an access point in the bibliographic records. If it is unique, it works like an identifier for the authority record, that contains the rest of the variants: the preferred form used as an access point links the bibliographic record to the authority record. In many recent catalogs,

the link from the bibliographic record to the authority record is not the preferred form but the authority number. It is the case in Sudoc. The Sudoc bibliographic record access points are just authority record identifiers. When the record is displayed or exported, the Sudoc system follows the link and substitutes a name or a term to the identifier. Sudoc has more than ten million bibliographic records and two million authority records. An authority record is not an entry in a biographical dictionary. It is supposed to contain nothing but information sufficient to distinguish a person from another one described by another authority record. Authority records for ppeople contain mainly information about their names, their dates and their country. Authority records for concepts contain information about their labels and their relationships to other concepts (broader, narrower, etc).

2.2 Semantization of Bibliographic Metadata

Reasoners need our records to be expressed in RDF. As RDF is nothing but a generic model, one have to choose one or several RDF vocabularies. This vocabulary had to meet some expectations:

- to be able to express precise bibliographic assertions
- to be minimally stable and maintained by a community

The FRBRoo vocabulary meets these expectations:

- It has been develpped for fine grained bibliographic requirements
- It is well documented
- It is maintained and still developed by an active community
- It is connected to other major modelling iniatives.

First, FRBRoo takes it main concepts form FRBR (1998), a prominent model that was developed by the International Federation of Library Associations during the 1990's. FRBRoo keeps core FRBR(1998) concepts but claims to overcome some of its alleged limitations. Second, FRBRoo is built as an extension of another model for cultural objects, namely CRM CIDOC. CRM CIDOC's main scope is material cultural heritage, as curated by art or natural history museums. It is focused on expressing the various events that constitute an object life, before or after its accession to the museum. FRBRoo imports many of its classes and properties from CIDOC CRM, but needs to forge a lot of new ones, to cope with more abstracts entities as texts and works.

For SudocAD's needs, it was not necessary to convert all UNIMARC fields in FRBRoo. But even for the conversion of the fields needed for the reasoning, we had to extend FRBRoo and forge some new classes and properties.

3 The Method

In a very general setting one has: The data: a base $\mathcal{D}$ of document notices, a base $\mathcal{A}$ of authority notices and a (new) document notice d. The bases are represented in a knowledge representation language based on a formal ontology $\mathcal{O}_B$.

The problem: link an entity E in d, representing an object O in the exterior world, to an authority notice in $\mathcal{A}$, representing the same object O, if such an authority is in $\mathcal{A}$.

Important tasks of the implemented method can be briefly stated as follows.

- Linkage knowledge: comparison criteria and rules.

  A fundamental task consists of building linkage knowledge whose main components are comparison criteria and logical rules.

  An *elementary comparison criterion* is a function c_δ that associates, with respect to a specific notion δ, a value to a couple composed of an entity E in d and an authority A in $\mathcal{A}$. For instance, a period criterion can use the publication date of d and, if A is a person authority, the life dates of A. For computing the value of a period criterion between an author E of d, and an authority, one can use knowledge such as "if the publication date of d is t then E cannot be identical to a person authority A whose birth date is posterior to t." The set of values of a comparison criterion is a totally ordered set of qualitative values representing the similarity between E and A with respect to the notion δ (typically: $similar, intermediate, weak, dissimilar$).

  The (global) comparison between an entity E and an authority A is also express as a qualitative value (e.g. $strong$ or $medium$, ..., or $impossible$) which is the conclusion of a logical rule having values of elementary comparison criterion as hypotheses (e.g. "if the value of the date criterion is $impossible$ then the global comparison has $impossible$ for value".) Note that computing values of criteria may need an alignment step between data in d and data in the A'_s.

  In order to precisely define comparison criteria one has to build a " linkage ontology" $\mathcal{O}_L$ which contains the concepts and the relations needed to express the comparison criteria.

- Working Knowledge Base

  Having defining comparison criterion, the next step consists of restricting and transforming the database into a knowledge base that only contains useful information for linking entities in d to authorities in $\mathcal{A}$. A piece of information may be useful for the linkage problem if it is used in a comparison criterion. Let $\mathcal{W}$ be this knowledge base which is based on the formal ontology composed of the ontology $\mathcal{O}_B$, in which $\mathcal{D}$ and $\mathcal{A}$ are expressed, and the linkage ontology $\mathcal{O}_L$. $\mathcal{W}$ should have two main properties: it should contain all the authorities in $\mathcal{A}$ which may be linked to entities in d and it should be small enough to do efficiently the computations needed by the linkage problems.

- Authority Enrichment

  The links between document notices and authority notices in $\mathcal{W}$ are used for enrichment of the authority notices. Enrichment of an authority notice A consists of specializing A. For instance, if an authority A is author of a lot of documents dealing with medicine it is probably relevant to add that medicine is within the competence of A. Applying this rule leads to adding a value to an existing attribute whenever $competence$ is initially in A , otherwise, if $competence$ is a linkage concept in $\mathcal{O}_L$ which is not in the initial ontology $\mathcal{O}_B$ it leads to adding a new attribute to A.

- Linkage Computations

  For each couple (E, A), where E is an entity in d which may be linked to an authority A, and for each comparison criterion c_δ one has to compute, in $\mathcal{W}$, the value $c_\delta(A, E)$. This computation may use numerical computations but the values are qualitative values (e.g., $similar, intermediate, weak, dissimilar$) so that can be used as hypotheses in logical rules.

  Finally, logical rules, having values of comparison criteria (between A and E) in hypothesis and a value of the global qualitative comparison criterion, $link$, in conclusion are fired. The result is a partition of the authority candidates ordered by decreasing relevance with respect to the possibility of linking E and A. For instance, if $link(E, A) = strong$ there is strong evidence that E and A can be linked, i.e. that they represent the same in the exterior world.

3.1 Working Base and authority candidates

3.1.1 Principles

In this paper we consider named entities. It means that considered objects have an important discriminant attribute which is their name. In d an entity E is thus identified by a name. It is also assumed that an authority in $\mathcal{A}$ has a specific attribute $denomination$, which is the set of different names used for the object represented by this authority.

The whole data base is very large and a first step is to limit it to useful information for the linkage problem. This working base should ideally contain all the possible candidates and should not be too large in order to be able to process it efficiently. As our problem is to link named entities in d to existing authorities having names in the authority notice base, we start by using a name criterion.

Thus, the first step consists of constructing a graph containing (potentially all) the possible candidates, information useful for the linkage problem and this graph being not too large in order to be able to process it efficiently.

3.1.2 In sudocAD

The working base is built as follows.

1. For each author name in d, the first task is to represent an author name in d in the same way, say $name$, as denomination in authority notices (this may need an alignment between 'author name' in d and 'denomination' in authority notices).

2. The set $\mathcal{A}' = sim(name)$ of authorities in $\mathcal{A}$ having a denomination, (either restraint or rejected if there is such a discrimination in the base), similar to $name$ is computed.

3. For each authority A in $\mathcal{A}'$, the set $Bib(A)$ of bibliographic notices having A as an author or as a significant role is computed. As we only consider named entities of type $Author$ in d, in $Bib(A)$ we only consider documents for which A is a contributor with a role $compatible$ with the role $Author$ (e.g. $ScientificEditor$, $PhDAdvisor$, ...).

4. The working base $\mathcal{W}(name)$ is obtained by making the union of the authority notices in A and the document notices $Bib(A)$ for all A in $\mathcal{A}'$.

A fundamental assumption is that the function sim is sufficiently robust to author name variations in order that if there is an authority in $\mathcal{A}$ corresponding to the author whose name is $name$ in d then this authority is (almost surely) in $\mathcal{A}'$. Another assumption is that $W(name)$ contains sufficient contextual knowledge concerning the authorities in order to remove the ambiguities, i.e. to solve the linkage problem. The linkage problem can be now stated as follows: for each $name$ of a named entity in d, compute a set of authority notices in $\mathcal{A}'$ ordered by decreasing relevance with respect to identity, using $\mathcal{W}$, d, the domain and the linkage ontologies and general knowledge represented by rules (cf.).

3.2 Super-authority

3.2.1 Principles

The aim of this step is to enrich the data concerning an authority A with information that can be used in comparison criteria. For doing that, information concerning documents in which A is a contributor are computed, synthesized and then attached to A.

3.2.2 In sudocAD

In d, which is a notice of a paper in a scientific journal, one has only, besides names of the authors, a publication date, the paper title, the language and a list of scientific domains. For each of this notions one can possibly enrich an authority notice by using information in the document notices, $Bib(A)$, in which A is a contributor. For instance, aggregating all domains of notices in $Bib(A)$ is, generally, more precise than a piece of information concerning the competence in A. In the same way, one can compute, then associate to A, an interval of publication dates. Note that, due to the nature of d (scientific paper), the kinds of contributor considered have been restricted to those having a scientific role, e.g., author, PhD supervisor, scientific editor, preface writer, etc. From now on, for simplicity sake, $\mathcal{A}$ denotes the set of super-authorities and a super-authority is simply called an authority.

3.3 Comparison Criteria

3.3.1 Principles

Let us call $dimension$ an abstract notion for which information exist both for named entities in d and in authorities. For instance, as seen previously, the time is such a dimension since we have a publication date in d and (possibly) life dates and a publication date interval in authorities. For each dimension δ, a comparison criterion c_δ is built that associates to a pair (E, A), where E is a named entity in d which may be linked to A, a qualitative value.

These values can be considered as propositional symbols for doing reasonings in the following way. Indeed, let us consider a pair (E, A) and a comparison criterion c_δ whose values are the set $v_1, \ldots, v_k$. Let us assume that $c_\delta(E, A) = v_i$ then considered as propositional symbols, v_i takes the value $true$ while the others take the value $false$ (cf. section 3.4 to see how they are used in rules) .

3.3.2 In sudocAD

The dimensions taken into account in sudocAD are: $\{denomination, domain, time, language\}$. The sets of values of their associated criteria contain from 2 to 4 values ordered with respect to the similarity relevance.

- Denomination

  The set of values of the 'denomination' criterion is : $\{sameDenomination, closeDenomination, distantDe$
  Given a name of an author in d and a denomination of an authority A, the sudocAD algorithm has two parameters: a threshold and an algorithm for computing an edition distance. For the evaluated sudocAD system, the threshold is 0.8 and the Levenshtein's algorithm is used.

  Split n_d, the name of an author E in d, and n_A, a denomination of an authority A, into two strings, respectively (n_1, p_1) and (n_2, p_2). The first string is the most discriminant part of the name, in our case it corresponds to the family name, and the second string, less important, is composed of first names or first name initials. The strings n_1, n_2, p_1, p_2 are normalized (transformation of uppercases into lowercases, deletion of accents and of redundant spaces, etc.). Two independent functions compare respectively n_1, n_2 and p_1, p_2. For both, their result is one of the same qualitative value set $identical, stronglyCompatible, compatible, distant, different$, they use the same threshold and the same distance edition algorithm, but due to their different nature(e.g., initials in p_i's) the two functions are rather different. For instance, they differently use the prefix notion.

  The results of these two functions then are aggregated as follows.

  if (n_1 and n_2 are identical or strongly compatible) then (if p_1 and p_2 are identical or strongly compatible return SAME else if p_1 and p_2 are compatible or distant return CLOSE else return DISTANT)

else if(n_1 and n_2 are compatible) then (if p_1 and p_2 are identical or strongly compatible or compatible return CLOSE else if p_1 and p_2 are distant return DISTANT else return DISSIMILAR)

else if(n_1 and n_2 are distant) then (if p_1 and p_2 are identical or strongly compatible or compatible or distant return DISTANT else return DISSIMILAR) return DISSIMILAR

Finally, as an authority A may have a set of denominations, $Denom(A)$, the value of the denomination criterion $c_{denomination}(E, A)$ is equal to the maximum value over the set of denominations of A, i.e., $c_{denomination}(E, A) = max\{c_{denomination}(n_d, n_A)|n_A \in Denom(A)\}$, where n_d is the name of E.

For more details see Annex.

- Domain

  The set of values of the 'domain' criterion is : $\{domainStrongCorrespondence, domainIntermediateCorr$
  We compare the set of domains of the Journal in which d has been published with the weighted list of domains in the authority A.

  More precisely, to each A a domain profile is associated. A domain profile is a set of weighted domain $\{(d_1, p_1), \ldots, (d_k, p_k)\}$ computed as follows. To a document which is in $Bib(A)$, i.e., for which A is a scientific contributor, is associated a set $\{(d_1, q_1), \ldots, (d_m, q_m)\}$ where $\{d_1, \ldots, d_m\}$ is the set of domains occurring in the document and a document counts for one, i.e.,

  $$\sum_{i=1}^{m} q_i = 1$$

  Such a set can be considered as a vector on the set of domains and the domain profile of A is the sum of these vectors. In the same way a domain profile can also be associated with the document d, the weight of a domain of d being equal to $1/\#dd$, where $\#dd$ is the number of domains associated with d. The profile domain of any entity in d is the domain profile of d.

  The similarity measure between two domain profiles $P = \{(d_1, p_1), \ldots, (d_m, p_m)\}$ and $P' = \{(d'_1, p'_1), \ldots, (d'_n, p'_n)\}$ that have been used in sudocAD is

  $$sim(P, P') = \sum_{i=1}^{m}\sum_{j=1}^{n} p_i p'_j \sigma(d_i, d_j) / \sum_{i=1}^{m}\sum_{j=1}^{n} p_i p'_j$$

  In this formula,

  $$\sigma(d, d') \in [0, 1]$$

  is a similarity between domains satisfying, $\sigma(d, d') = 1$ means that $d = d'$ or that d and d' are synonyms, $\sigma(d, d') = 0$ means that it is quite impossible that a given person can have a scientific role in a document about d and a document about d'.

  The qualitative values for the domain criterion are defined as follows, $c_{domain}(d, A)$ is equal to:

  - $domainStrongCorrespondence$ whenever $0.8 < sim(P, P') \leq 1$,
  - $domainIntermediateCorrespondence$ whenever $0.5 < sim(P, P') \leq 0.8$,
  - $domainWeakCorrespondence$ whenever $0.2 < sim(P, P') \leq 0.5$,
  - $domainWithoutCorrespondence$ whenever $0 \geq sim(P, P') \leq 0.2$.

  Note that we take $c_{domain}(E, A) = c_{domain}(d, A)$ for any entity in d.

  For more details see Annex.

- Date

  The set of values of the 'date' criterion is : $\{dateStrongCorrespondence, dateIntermediateCorrespondenc$
  An overview of the date criterion is as follows. Two booleans are used $compatiblePeriod$ expressing compatibility between the publication date p_d of d and the interval of publication dates $(beginPeriod, endPeriod)$ of A and $compatibleLife$ expressing compatibility between the publication date of d and the life interval $(birthDate, deathDate)$ of A. If the birth date of the death date of A does not exist then they are given the value $null$. There are three thresholds: T_1 for the age at which a person can publish (set to 20 in sudocAD), T_2 for the length life whenever either the birth date or the death date of an author is unknown (set to 100 in sudocAD) and T_3 (set to 10 in sudocAD) to express that if the publication date of d is within the interval of publication dates plus or minus T_3 then they are compatible.

  compatiblePeriod= $(beginPeriod \leq datePubli)$ and $(endPeriod \geq datePubli)$;

  if ($birthDate \neq null$ and $deathDate = null$) then $deathDate = birthDate + T_2$;

  if ($deathDate \neq null$ and $birthDate = null$) then $birthDate = birthDate - T_2$;

  if($birthDate \neq null$ and $p_d \geq birthDate{+}20$ and $p_d \leq deathDate$) then $compatibleLife = true$;

  if($birthDate \neq null$ and $birthDate + 20 > p_d$) then return $dateWithoutCorrespondence$;

  if(compatiblePeriod and compatibleVie) then return $dateStronCcorrespondence$;

  if(compatiblePeriod or compatibleVie) then return $dateIntermediateCorrespondence$;

  $compatiblePeriod = (beginPeriod \leq p_d)$ and $(endPeriod + 10 \geq datePubli)$;

  if(compatiblePeriod and ($birthDate = null$ or $birthDate{+}20 \leq p_d$)) then return $dateWeakCorrespondence$

  if(($birthDate = null$) or $birthDate + 20 \leq datePubli$) then return $dateWeakCorrespondence$;

  For more details see Annex.

- Language

  The set of values of the 'language' criterion is : $\{languageStrongCorrespondence, languageWithoutCorr$
  For our experiment the language is not discriminant and we have chosen a simplistic language criterion: if in $Bib(A)$ there is a document written in the same language as d then the value of $c_{language}(d, A)$ is $language_strong_correspondence$ otherwise it is $language_without_correspondence$. Note that, as well as for the domain criterion, the language criterion deals with d and is transferred to each entity E in d.

  For more details see Annex.

3.4 Linkage

3.4.1 Principles

Let E be an entity in d and $\mathcal{A}$ the set of authorities. For any criterion c_δ we have the value $c_\delta(E, A)$ of this criterion for the pair (E, A). These values are used for computing the possibility of linkage between E and A. This possibility is expressed by a (global) comparison criterion called $linkage$. As well as for the elementary comparison criteria the values of are qualitative values, e.g. $strongLinkage$ or $impossibleLinkage$ and they can be considered as propositional symbols. Let us assume that there is k dimensions $\delta_1, \ldots, \delta_k$.

The hypothesis of a rule is a conjunction of k propositional symbols, H_1 and ... and H_k, where for all i H_i is a value of c_{δ_i}. The conclusion of a rule is a value $linkage(E, A)$ of the linkage criterion for the pair (E, A). For each value of $linkage$ we have a set of rules having this value as conclusion. Here is an example of a rule. If $sameDenomination$ and$dateIntermediateCorrespondence$ and

$domainStrongCorrespondence$ and $languageStrongCorrespondence$ then $strongLinkage$. This rule is used as follows. If for a given pair (E, A) all hypotheses are true then the system concludes that there is a strong evidence that E and A represent the same entity. Said otherwise, the system cannot distinguish E from A. At the opposite, if $linkage(E, A) = impossibleLinkage$ it means that the system considers that there is strong evidence that E and A do not refer to the same exterior object.

The set of rules are used to partition the authorities. If the values of $linkage$ are $\{v_1, \ldots, v_n\}$ then there is at most n classes defined by the value of $linkage(E, A)$, i.e. the i-th class is the set $\{A \in \mathcal{A} | linkage(E, A) = v_i\}$.

The results will be used differently depending on whether the system is used in an automatic mode or as an aid to a human operator. The choices made in sudocAD for these two modes are presented in section 4.

3.4.2 Implementation in sudocAD

Seven qualitative values have been considered for the linkage criterion: $\{strongLinkage, mediumLinkage, weakL$

The rules are described below in an array.The lines correspond to rules and the columns correspond to the comparison criteria. The following notations are used. The values of the elementary comparison criteria are represented as follows.

- values of the denomination criterion: +++, ++, +, - (where: +++ stands for $sameDenomination$, ++ for $closeDenomination$, + for $distantDenomination$, - for $dissimilarDenomination$)
- values of the date criterion: +++, ++, +, ?, - (+++ stands for $dateStrongCorrespondence$, ++ for $dateIntermediateCorrespondence$, + for $dateWeakCorrespondence$, - for $dateWithoutCorrespondenc$ and ? for unknown)
- values of the domain criterion: +++, ++, +, ?, - (+++ stands for $domainStrongCorrespondence$, ++ for $domainIntermediateCorrespondence$, + for $domainWeakCorrespondence$, - for $domainWithoutC$ and ? for unknown)
- values of the language criterion: +, ?, - (+ stands for $languageStrongCorrespondence$, - for $languageWithoutCorrespondence$)

The 300 cases are gathered into 7 subsets labeled S, M, W, P, N, U, I correspond respectively to the rules concluding by $strongLinkage, mediumLinkage, weakLinkage, poorLinkage, neutralLinkage, unrelatedLink$

Linkage	Denomination	Date	Domain	Language	Nb
S					**3**
LS1	$\{+++\}$	$\{+++\}$	$\{+++, ++\}$	$\{+\}$	2
LS2	$\{+++\}$	$\{++\}$	$\{+++\}$	$\{+\}$	1
M					**15**
LM1	$\{+++\}$	$\{+, ?\}$	$\{+++\}$	$\{+\}$	2
	$\{+++\}$	$\{+++, ++, +, ?\}$	$\{+++\}$	$\{?\}$	4
LM2	$\{+++\}$	$\{++, +\}$	$\{++\}$	$\{+\}$	2
LM3	$\{++\}$	$\{+++\}$	$\{+++\}$	$\{+, ?\}$	2
LM4	$\{++\}$	$\{++\}$	$\{+++, ++\}$	$\{+\}$	2
	$\{++\}$	$\{+++\}$	$\{++\}$	$\{+\}$	1
LM5	$\{+++\}$	$\{+++, ++\}$	$\{+\}$	$\{+\}$	2
W					**23**
LW1	$\{+++\}$	$\{+++, ++\}$	$\{++, +\}$	$\{?\}$	4
	$\{++\}$	$\{++\}$	$\{+++, ++, +\}$	$\{?\}$	3
	$\{++\}$	$\{+++\}$	$\{++, +\}$	$\{?\}$	2
	$\{++\}$	$\{+++, ++\}$	$\{+\}$	$\{+\}$	2
LW2	$\{++\}$	$\{+\}$	$\{+++, ++\}$	$\{+\}$	2
LW3	$\{+\}$	$\{+++\}$	$\{+++\}$	$\{+, ?\}$	2
LW4	$\{+++\}$	$\{+\}$	$\{+\}$	$\{+\}$	1
	$\{+++\}$	$\{+\}$	$\{++, +\}$	$\{?\}$	2
	$\{++\}$	$\{+\}$	$\{+\}$	$\{+\}$	1
	$\{++\}$	$\{+\}$	$\{+++, ++, +\}$	$\{?\}$	3
LW5	$\{+++\}$	$\{?\}$	$\{++\}$	$\{+\}$	1
P					**119**
LP1	$\{+++\}$	$\{+++, ++, +, ?\}$	$\{-\}$	$\{+, ?\}$	8
LP2	$\{+++\}$	$\{?\}$	$\{+, ?\}$	$\{+, ?\}$	4
	$\{+++\}$	$\{?\}$	$\{++\}$	$\{?\}$	1
	$\{+++\}$	$\{+++, ++, +\}$	$\{?\}$	$\{+, ?\}$	6
	$\{++\}$	$\{?\}$	$\{+++, ++, +, ?\}$	$\{+, ?\}$	8
	$\{++\}$	$\{+++, ++, +\}$	$\{?\}$	$\{+, ?\}$	6
	$\{++\}$	$\{+++, ++, +, ?\}$	$\{-\}$	$\{+, ?\}$	8
LP3	$\{+\}$	$\{+++\}$	$\{++, +, ?, -\}$	$\{+, ?\}$	8
	$\{+\}$	$\{++\}$	$\{+++, ++, +, ?, -\}$	$\{+, ?\}$	10
LP4	$\{+++, ++, +\}$	$\{+++, ++, +, ?\}$	$\{+++, ++, +, ?, -\}$	$\{-\}$	60
N					**18**
Another case	$\{+\}$	$\{+, ?\}$	$\{+++, ++, +, ?\}$	$\{+, ?\}$	16
	$\{+\}$	$\{?\}$	$\{-\}$	$\{+, ?\}$	2
U	pb. LP4				**38**
LU1	$\{\}$	$\{\}$	$\{\}$	$\{-\}$	0
LU2	$\{+\}$	$\{+\}$	$\{-\}$	$\{+, ?\}$	2
LU3	$\{+++, ++, +\}$	$\{-\}$	$\{+++, ++, +, ?\}$	$\{+, ?, -\}$	36
I					**84**
LI1	$\{+++, ++, +, -\}$	$\{-\}$	$\{-\}$	$\{+, ?, -\}$	12
LI2	$\{-\}$	$\{+++, ++, +, ?\}$	$\{+++, ++, +, ?\}$	$\{+, ?, -\}$	48
	$\{-\}$	$\{-\}$	$\{+++, ++, +, ?\}$	$\{+, ?, -\}$	12
	$\{-\}$	$\{+++, ++, +, ?\}$	$\{-\}$	$\{+, ?, -\}$	12

Actually, we have built only 24 rules used in a specific way which lead to a result identical as the one just described. This is explained in section 4.

4 The System SudocAD

(to be finalized)

The system sudocAD has been developed upon COGUI. COGUI is a platform for designing, building and reasoning on conceptual graph knowledge-bases (see [**?**]) that has been used at each stage of the project. The project is persistently stored in the xml file Xblablabla. We will briefly browse through the project to get an idea of what sorts of functionality are provided. In order to fully understand the capabilities of COGUI one has to read the documentation.

- *The Ontology* First thing to do is to store on disk the project file then download COGUI and open the file. By clicking on the vocabulary tab on the vertical left panel it is possible to see the ontology used. It contains a hierarchy of (313) concept types and a hierarchy of (1179) relation types (nesting types and modules are not used in sudocAD). A tree view is presented in the left vertical panel and a graphical view in the top right panel (the cycles are due to the existence of synonyms between notions in the standards cidoc-crm and frbr). An entity may have several types and banned types (bottom left panel) express incoherences, for instance there can not exist an entity which would be a $PhysicalThing$ and a $ConceptualObject$. The relation types are also hierarchically organized. Note that in sudocAD there are only binary relations but relations with any arity can be used in COGUI. The root of the linkage relations is emphliage : binding vocabulary relation (Resource,Resource) at the bottom of the relation types panel. A relation has a signature which indicates its arity and the maximal types an attribute can have (for instance the two attributes of the relation emphliage : liageAuthority (Person,Person) have to be of type (less than or equal to) Person. Patterns and Prototypes are not used in sudocAD.

- *Input Data* Input data consist of a part of the sudoc database and of a PERSEE notice. As explained in section 2 the sudoc database have been translated into RDF. A part of it, containing authority notices whose denominations are closed to the name of an author in the PERSEE notice and the bibliographic notices in which these authorities have a scientific role, is imported into cogui through the import RDF/S natural mode tool. The file Xblablabla contains ...

- *The scripts* The method described in section 3 is implemented through the scripts which use the queries for searching the data graph. We will skim through the "STEP_BY_STEP" mode which allows to understand how the system works.

  - *STEP1*
  - *STEP2*
  - *STEP3*
  - *STEP4*
  - *STEP5* We give here the 24 rules in STEP5 scripts and the way they are actually used in sudocAD. Their semantics is equivalent to the 300 rules previously given.

Rule	Denomination	Date	Domain	Language	Linkage
LS1	+++	+++	++	+	S
LS2	+++	++	+++	+	S
LM1	+++	*	+++	*	M
LM2	+++	+	++	+	M
LM3	++	+++	+++	*	M
LM4	++	++	++	+	M
LM5	+++	++	+	+	M
LW1	++	++	+	*	W
LW2	++	+	++	+	W
LW3	+	+++	+++	*	W
LW4	++	+	+	*	W
LW5	+++	*	++	+	W
LP1	+++	*	-	*	P
LP2	++	*	*	*	P
LP3	+	++	*	*	P
LP4	*	*	*	-	P
LU1	+	*	*	-	U
LU2	*	+	-	*	U
LU3	*	-	*	*	U
LI1	*	-	-	*	I
LI2	-	*	*	*	I
Other case	*	*	*	*	N

These rules are fired in a specific order and for a given pair (E, A) the value $linkage(E, A)$ is the first value obtained. As soon as a rule is fired for a pair (E, A) the others are not fired. The chosen order, which is important for the result is as follows: LI1, LI2, LU3, LP4, LS1, LS2, LM1, LM2, LM3, LM4, LM5, LW1, LW2, LW3, LW4, LW5, LP1, LP2, LP3 LU1, LU2, Another case.

The order on the set of values of a comparison criterion is also used. If the value of a criterion is positive, say p, then it is assumed that it has also all the positive values $p' \leq p$. Note also that the joker * stands for any value of a criterion as well as the absence of value.

- *STEP6*
- *STEP7*

5 Evaluation

5.1 Methodology

Given a base $\mathcal{D}$ of document notices, a base $\mathcal{A}$ of authority notices and an entity E in a document notice d results of sudocAD is essentially a partition of $\mathcal{A}$ in classes ordered by decreasing relevance with respect to co-reference. It seems difficult to assess the quality of this partition and even to define what could be the quality of such a partition. Thus, we consider two ways of using this partition which can be evaluated by human experts: an automatic mode and a decision-aided mode. In the automatic mode, the system either proposes an authority A to be linked to E or proposes no link. In the decision-aided mode, the system proposes a list of authorities in a decreasing relevance order until the operator chooses one authority or stops using the system.

From the database PERSEE, 150 bibliographic notices, referencing 212 authors, have been chosen at random. For these notices, professional librarians had to do their usual work, that is to try to link authors in PERSEE notices to authorities, and that in their usual work environment. That is to say, librarians had the complete PERSEE notices (even if parts of the notices are not taking into account by sudocAD, for instance the title of the paper) and on-line access to the bases $\mathcal{D}$ and $\mathcal{A}$. Librarians had also a limited time, no more than 5 minutes for linking an author, and they also have to respect the usual constraint to not create erroneous links. For an author E in a PERSEE notice a librarian could take one of the following decisions:

- link with certainty E to an authority A
- link with uncertainty E to an authority A and suggest other possible authorities
- refrain for linking with certainty
- refrain for linking with uncertainty nevertheless suggest possible authorities

A librarian analyzed the results of this first step. He was not involved in this step and did not work in the usual work environment a librarian has for linking because he could use any sources of information (e.g. the web) and had no limited time. After having modified, if necessary, results of the first step, it is hoped to have the best possible linkages to compare with those obtained by sudocAD.

Expert Results	**Number**
Linkage with certainty	146
Linkage without certainty and other choices	3
Linkage without certainty but no other choices	19
No linkage with certainty	37
No linkage without certainty but choices	7
No linkage without certainty and no other choices	0

5.2 Automatic Linkage

We considered four different ways for automatic linkage listed as follows for the most restrictive to the less restrictive.

- AL_1 If the best class, i.e. $strongLinkage$, contains only one authority then E is linked to this authority

- AL_2 If the union of the two best classes, i.e. $strongLinkage$ and $mediumLinkage$, contains only one authority then E is linked to this authority
- AL_3 If the union of the three best classes, i.e. $strongLinkage$ and $mediumLinkage$ and $weakLinkage$, contains only one authority then E is linked to this authority
- AL_4 If the union of the four best classes, i.e. $strongLinkage$ and $mediumLinkage$ and $weakLinkage$ and $poorLinkage$, contains only one authority then E is linked to this authority

For the comparison between the expert choice and one of these methods we consider that the answer given by the method is a

- *Good decision* either when the expert links with certainty E to A as well as the method or when the expert does not link with certainty E and the method does not propose a link
- *Acceptable decision* either when the expert links with uncertainty E to A and the method links E to A or when the expert does not link with certainty E and the method proposes a link to a doubtful candidate
- *Bad decision* either when the expert links with certainty E to A and the method links E to $A' \neq A$ or when the expert does not link with certainty E and the method proposes a link
- *Prudent decision* when the expert links with or without certainty and the method does not propose a link

The means of these parameters for the 212 authors occurring in the 150 PERSEE bibliographic notices are as follows.

Method	Good decision	Acceptable decision	Bad decision	Prudent decision
AL_1	54.7%	0%	1.89%	43.4%
AL_2	77.36%	0.47%	1.89%	20.28%
AL_3	80.19%	0.47%	3.77%	15.57%
AL_4	86.79%	0.94%	6.6%	5.66%

5.3 Decision-Aided

In a decision-aided mode the system presents to a human operator an ordered list of the candidate authorities. This list is presented in the decreasing order of relevance $strongLinkage$, $mediumLinkage$, $weakLinkage$, etc. until the operator chooses one authority to be linked to E or stops the unfolding of the candidate authorities and concludes that there is no authority which can be linked to E.

Three classical parameters in Information Retrieval have been considered for evaluating the use of our system in a decision-aided mode. They use the following sets of authorities:

- $Candidates$ is the set of authority candidates in the working base
- $Impossible$ is the set of authorities related by $impossibleLinkage$ to E
- $Selected$ is the union of the authority linked to E by the operator and, when there is uncertainty, the set of possible authorities proposed by the operator

The parameters are then defined as follows.

- *Recall*

  $recall = |Selected \cap (Candidates \setminus Impossible)|/|Selected|$

- *Precision*

  $precision = |Selected \cap (Candidates \setminus Impossible)|/|(Candidates \setminus Impossible)|$

- *Relevance*

  $relevance = |Selected/MaxPos|$,

  where $MaxPos$ is the last position in the ordered result of an authority in $Selected$.

There are two different situations: the operator can decide either to link or not to link E to an authority . If he decides to link the recall is not relevant, because in this case the recall is equal to 1 unless $Selected \cap Impossible \neq \emptyset$ which is unlikely. If he decides not to link *Selected* is empty. In this case another parameter could be $|Impossible|/|Candidates|$ if $Candidates$ is not empty, this is generally the case since the construction of $Candidates$ is based on the denomination.

The means of these parameters for the 212 authors occurring in the 150 PERSEE bibliographic notices are as follows.

Expert choice	Recall	Precision	Relevance
No linkage with uncertainty	100%	45.41%	60.71%
Linkage with certainty	100%	77.57%	94.32%
Linkage without uncertainty	100%	68.01%	95.24%

Note that this is only a shallow evaluation that should be deepened. Indeed, the measure of performance of a decision-aided system is the gain obtained when the operator uses the system compare to not using the system. But, the linkage problem cannot be solved without a system (unless proposing at random candidate authorities ...). So, for a significant evaluation one should make an experiment comparing two systems. This is a further work that is planned for a decision-aided use of our system but also for an automatic mode. .

6 Conclusion and further work

(to be finalized)

SudocAD deals with authors but can be used for other authorities as well, e.g. collective entities. The algorithms used in SudocAD assume that the bases are correct. If each notice, either document notices or authority notice, can be assumed to be correct links between them can be erroneous. Further work will take into account the quality of the considered bases.

References