


Hunting Ancestors: A Unified Approach for Discovering Genealogical Information

José João Almeida 

Centro Algoritmi, Universidade do Minho, Portugal

Departamento de Informática, Campus de Gualtar, Universidade do Minho, Portugal

jj@di.uminho.pt

Rui Castro Mendes 

Centro Algoritmi, Universidade do Minho, Portugal

Departamento de Informática, Campus de Gualtar, Universidade do Minho, Portugal

azuki@di.uminho.pt

Abstract

This paper presents an unified approach for discovering genealogical information. It presents a frameworks for storing information concerning ancestors, locations, dates and documents. It also intends to provide a framework that is able to perform inference concerning dates by using constraints and for handling relations, locations and sources. The DSL presented also aims to help users store information from heterogeneous sources along with the evidence contained therein.

2012 ACM Subject Classification Software and its engineering → Domain specific languages; Software and its engineering → Scripting languages; Theory of computation → Constraint and logic programming

Keywords and phrases Genealogy, Domain Specific Language, Temporal Constraints

Digital Object Identifier 10.4230/OASICS.SLATE.2019.22

Funding This work has been supported by FCT – Fundação para a Ciência e Tecnologia within the Project Scope: UID/CEC/00319/2019.

1 Introduction

In the area of digital humanities it is very important to register notes concerning people, genealogy, documents, news, photos, houses, family histories, etc. However, gathering information concerning ancestors is hard. Information is often incomplete including chronological information, lack of records or records containing incomplete or unreadable information and also photographs depicting people or places unknown. Depending on the timeframe or locale, there can be many sources for gathering information. These sources may include parish records, newspapers, and records concerning commerce or school. More recently, there can also be information in photographs. These often involve events where the family took photographs with a large assembly of its members and these photographs can support evidence of family ties that can be hard to find because the location of the photograph and many of the people therein can be unknown.

1.1 Motivation

The goal of this paper is to systematize information concerning sources, evidence and chronology. This will be performed by:

- adding reasoning concerning chronology, like probable dates of birth, christening, attending school, immigration, opening practice and death.
- having information concerning online sources for records and publications including newspapers



© José J. Almeida and Rui C. Mendes;

licensed under Creative Commons License CC-BY

8th Symposium on Languages, Applications and Technologies (SLATE 2019).

Editors: Ricardo Rodrigues, Jan Janoušek, Luís Ferreira, Luísa Coheur, Fernando Batista, and Hugo Gonçalo Oliveira; Article No. 22; pp. 22:1–22:6



Open Access Series in Informatics

Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

- having information concerning documents, their OCR, evidence collected and inference
- having information concerning inference, namely concerning dates, probable locations (e.g., birth, marriage, death), probable ancestors or descendants and family ties

1.2 Information

If we know that a given ancestor posted an ad in a newspaper advertizing a practice as dentist and warning against other practitioners who do not have any sort of education, one can conclude that that ancestor must have some studies in a medical school, and, if he is opening at office, it is probable he is somewhat young.

We want a system were we can add information that could be used to infer interval bounds concerning dates of birth and death, that is capable of providing clues concerning sites where one can search for information (e.g., specialized schools and universities where our ancestor could have learned dentistry).

1.3 Chronology

Information providing hints concerning chronology can be obtained from many sources. In the example of the ad posted in the newspaper, it is possible to infer an interval for that person's date of birth (e.g., it was posted by someone who is starting practice and thus that person must be under forty at the time). This information often only provides us with guesses in the form of interval bounds but provides clues that, when combined with other evidence, can pinpoint chronological information [7, 3, 6].

1.4 Ancestors knowledge base

Registering genealogical information is a challenging task. In order to take sound genealogical notes we are trying to follow an ontology approach. We want to be able to:

1. Define the base concepts (classes), their expected data; properties and the concept taxonomy (by default person, families, photos, places, etc);
2. Define the relations (inverse, domain, range, properties);
3. Populate the genealogy with persons, families, houses, dates, etc;
4. Query using partial information in order to better understand the facts present in documents;
5. Build a versatile reasoner capable of solving constraints about family relations, date intervals, names, etc;
6. Build a name module capable of following human conventions and intuitions;
7. Provide context-aware clues about complementary information sources.

In section 2 we will discuss the ancestor notebook (*ANB*) language to help populate the genealogy. In section 3 we will discuss constraints and inference about ancestors data. Finally, section 4 will present the conclusions and future work.

2 Ancestors notebooks

We will present and discuss an ancestors notebook (*ANB*) DSL – a textual document where we intent to register genealogical information using a simple wiki-inspired domain specific language [2, 1].

ANB are used to register information about ancestors to be incorporated in our Ancestor Knowledge Base. This information can be seen as an ontology over person concepts like persons, families, houses, places, time, several types of documents, photos, portraits, organizations, and their relations.

ANB include textual documents with inline formal tuples (frequently triples, data properties) using the following structure:

In practical situations, we create a set of ANBs covering different family branches, archives, funds, albums, etc. During the creation process, we often consult previously stored knowledge. We illustrate this by presenting a small extract of an ANB:

```

#/Eduardo Honório de Lima           // main topic
#= Honório de Lima                   // alias
*1856 +1939

#doc HL1 {                           // related document
= Honório de Lima street, Porto

- O capitalista portuense #[Eduardo Honório da Lima] (1856-1939),
  colecionador de arte e grande amador de música.
  ( toponymy Archive of Porto)
}

#B Teotónio Augusto de Lima         // brother

#in-photo DA2-33                     // present in photo DA2-33
#dono fábrica de Cortumes do Bessa  // triples
#mecenas_of Museu Soares dos Reis  //
#lived_in casa HL1
#in-doc EAG1

===
#/photo:DA2-33                       // main topic is photo DA2-33
* c1890                               // taken in circa 1890
#in-album DiasA2                     // from album ...
#file f-33.jpg                       // filename ...
#desc { family of Honório de Lima .... } // textual description

===
#/house:casa HL1                     // main topic is house HL1
#= Moradia Honório de Lima I         // alias
#address Rua Cedofeita 492 a 498     // textual relation

- reconstructed em 1910
- it includes a small musical auditorium

```

A ANB can then be added (conciliated) with our ancestors knowledge base (AKB). First we can calculate the list of new instances – to check for typos (`anb-new-items myfamily.ab f.anb` command). And finally commit the new notes: (`anb-commit myfamily.ab f.anb` command). The commit process can produce warnings and errors when constraint violations are detected.

In a simplified way, `anb-commit` extracts triplets and subdocuments from the ANB and adds them to the AKB. There is also a simple script – `anb2triples f.anb` to extract a CSV version. This is the usual way of populating the ontology [4].

3 Inference

3.1 Example

Suppose we find a photograph with the portrait of a gentleman in his twenties. When we turn this photograph, we see the inscription “Jaime, irmão do tio Frederico”. The logo and address in the photograph indicates that it was taken by a photographer who worked in that specific address from 1929 to 1935. This photo belonged to an aunt called Blandina Neves. We can summarize the following information:

- This photograph is of a person whose name contains Jaime as a given name;
- This person has a brother called Frederico;
- Frederico is an uncle of the person who wrote the inscription in the back of the photograph;
- Jaime is apparently in his twenties;
- The photograph was taken between 1929 and 1935.

The normal constraints about people longevity, age, people-related intervals (cf. section 3.4) also stand. Using this system, we may write the query:

```
Jaime brother_of Frederico, Frederico uncle_of Blandina Neves,
Jaime:20..30 in_photo F1, F1 photo_taken 1929..1935.
```

This query is rewritten by our system into Prolog:

```
person(P1), name_includes(P1, 'Jaime'),
person(P2), brother_of(P1, P2), name_includes(P2, 'Frederico'),
person(P3), uncle_of(P2, P3), name_includes(P3, 'Blandina Neves'),
photo(F1), in_photo(P1, F1), apparent_age(F1, P1, 20..30),
photo_taken(F1, 1929..1935).
```

`person` is a fact that stores all information concerning a given person. This information can be used by other predicates like `brother_of` or `uncle_of` for checking for family ties or `name_includes` for checking the name of a person. `photo` stores all the information concerning a photograph including the people therein, location and when the photo was taken. When mentioning that a given person appears in a photo, it is possible to add information concerning the apparent age of the person.

Knowing when the photo was taken and the apparent age of the individual, it is possible to provide an estimate of the birth date of Jaime using finite domains [5] and, given that Jaime and Frederico are brothers, of also providing an estimate on his birth date and thus narrow the search scope.

3.2 Dealing with incomplete information

It is possible that the system does not have enough information to answer our query. When this happens, the query concerning, e.g, a person or a photograph, will return a placeholder that will be filled with unknown information. For instance, it is possible that we do not have information concerning Jaime but even so be able to find out who Frederico is. In this case, the system creates a record with an unknown name, `inf..sup` information for dates of birth and death, unknown location, etc. Then the system instantiates Jaime to be one of the names and subsequently constraints the date of birth knowing that the subject was between 20 and 30 years old somewhere between 1929 and 1935, yielding the interval 1899..1915.

3.3 Dealing with replicated information

During the research for information concerning relatives, it is possible to have more than one record concerning a given person. If this happens, the system should run a query that takes two profiles and computes their similarity. The similarity depends on instantiation. This is performed by comparing names, birth and death dates and data concerning other events, e.g., relations, locations and occupation. The name comparison is performed by performing a name similarity procedure (e.g., it could involve computing the number of names in common and their relative order). If two individuals are deemed similar, the system can show both records to the user and suggest that they may be merged. If the user input is positive, those records are merged.

3.4 Defining rules

It is also possible to define rules concerning constraints regarding relations. It is also possible to define properties regarding relations. These rules will be used for creating relations automatically. For instance, sibling is symmetric and transitive. Thus, if A and B and A and C are siblings, the symmetric and transitive closures of these relations are also created automatically. Furthermore, it is possible to establish constraints that will be applied using constraint propagation [5] for shrinking the time intervals concerning birth or death of people because of family relations.

We present some examples of the system. We state that two siblings cannot be born more than 30 years apart, that a person can only be mother of another with more than 15 years but less than 50 and that it must be alive at the time of birth. If a person appears in an active event, it must be older than 10 and must be alive for the event to take place. If a photo is taken and a person has a given apparent age, then we have more information concerning the person's birth date.

```

spouse: symmetric, anti-reflexive
sibling: transitive, symmetric, anti-reflexive
mother: anti-transitive, anti-symmetric, anti-reflexive
father: anti-transitive, anti-symmetric, anti-reflexive
person(P) :- P.death >= P.birth, P.death - P.birth <= 100
sibling(A,B) :- abs(A.birth - B.birth) < 30
mother(M, C) :- female(M), C.birth - M.birth > 15,
    C.birth - M.birth < 50, M.death >= C.birth
father(F, C) :- male(F), C.birth - F.birth > 15,
    C.birth - F.birth < 65, F.death + 1 >= C.birth
proof_of_life(E, P):- P.birth <= E.year, P.death >= E.year
is_active(E, P):- P.birth + 10 < E.year, P.death >= E.year
apparent_age(F, P, Apparent) :- (F.year - P.birth) in Apparent

```

3.5 Clues about information sources

Another important information in “hunting ancestors” is what information sources exist: these can contain clues of where, when and what to look for to complement current information about someone.

- ▶ **Example 1.** We know that someone born in Porto in 1870 is a dentist.
- The next step could be consulting the anuário of University of Coimbra of (years between 1890..1900) and the registers of Medical-Surgical Academy of Porto (1836-1911).
- Google(“Name” site:repositorio-tematico.up.pt)

► **Example 2.** We know that someone emigrated to Brazil, Rio Grande do Sul in 1829.
 → The next step could be consulting the “Lista das cartas de liberdade dos escravos Rio Grande do Sul” (Portalegre, Pelotas, RGS)
 → <http://www.apers.rs.gov.br/arquivos/>¹

In the current version, this module is in a very initial stage. They can be modelled with rules, in a similar way of what was done in section 3.4.

4 Conclusions

Although in a initial stage, this work already supports the importance of:

- having a constraint solver engine and versatile inference processes;
- providing a simple way of adding relations, their properties and related constraints;
- having domain specific languages for knowledge base population tasks;
- having a sound way of handing name similarity, alias and partial name unification;
- having simple queries dealing with partial names and family kinship (Luísa, sister of Margarida; tio Frederico; avô João).

Concerning future work, we aim to expand this system by creating a better name unification and similarity procedure, expand the DSL to better handle the information necessary and testing the system.

References

- 1 Tomaz Kosar, Sudev Bohra, and Marjan Mernik. Domain-specific languages: a systematic mapping study. *Information and Software Technology*, 71:77–91, 2016.
- 2 Tomaz Kosar, Nuno Oliveira, Marjan Mernik, Varanda João Maria Pereira, Matej Črepinšek, Cruz Daniela Da, and Rangel Pedro Henriques. Comparing general-purpose and domain-specific languages: An empirical study. *Computer Science and Information Systems*, 2010.
- 3 Philippe Laborie and Jerome Rogerie. Reasoning with Conditional Time-Intervals. In *FLAIRS conference*, pages 555–560, 2008.
- 4 J.B. Lamy. Owlready: Ontology-oriented programming in Python with automatic classification and high level constructs for biomedical ontologies. In *Artificial Intelligence In Medicine*, volume 80, pages 11–28, 2017.
- 5 Markus Triska. The Finite Domain Constraint Solver of SWI-Prolog. In *FLOPS*, volume 7294 of *LNCS*, pages 307–316, 2012.
- 6 Dongrui Wu and Jerry M. Mendel. Uncertainty measures for interval type-2 fuzzy sets. *Information Sciences*, 177(23):5378–5393, 2007. Including: Mathematics of Uncertainty. doi:10.1016/j.ins.2007.07.012.
- 7 Ran Zhao, Quang Xuan Do, and Dan Roth. A robust shallow temporal reasoning system. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Demonstration Session*, pages 29–32. Association for Computational Linguistics, 2012.

¹ Download and grep http://www.apers.rs.gov.br/arquivos/1169142561.Cat_Sel_Cartas_Liberdade_Vol_1.pdf