Extracting and Organizing Facts of Interest from OCRed Historical Documents

Joseph S. Park and David W. Embley
Brigham Young University, Provo, Utah 84602, U.S.A.

Abstract. Historical documents contain facts that family history enthusiasts are interested in extracting. In addition to fact extraction, organizing these facts into disambiguated entity records is also of interest. This paper shows how facts from an excerpt of a page in an OCRed book can be gathered automatically with some expert knowledge.

1 Introduction

Historians, genealogists, and others have great interest in extracting facts about persons and places found in historical documents and organizing these facts into disambiguated entity records. Figure 1, for example, shows part of a page from *The Ely Ancestry* [BEV02]. Facts of interest include those explicitly stated such as *William Gerard Lathrop was born in 1812*, *married Charlotte Brackett Jennings in 1837*, and *is the son of Mary Ely*. In addition to explicitly stated facts, implicit facts are also of interest. These include the fact that *William Gerard Lathrop has gender Male*, inferred from the stated fact that he is a son, and *Maria Jennings has surname Lathrop*, inferred from cultural tradition and the stated fact that her father has the surname Lathrop. Implicit facts also include disambiguating references to objects: the first and third Mary Ely mentioned are the same person, but not the same person as the second Mary Ely mentioned.

243311. Abigail Huntington Lathrop (widow), Boonton, N. J., b. 1810, dau. of Mary Ely and Gerard Lathrop; m. 1835, Donald McKenzie, West Indies, who was b. 1812, d. 1839.
(The widow is unable to give the names of her husband's parents.) Their children:

1. Mary Ely, b. 1836, d. 1859.
2. Gerard Lathrop, b. 1838.

243312. William Gerard Lathrop, Boonton, N. J., b. 1812, d. 1882, son of Mary Ely and Gerard Lathrop; m. 1837, Charlotte Brackett Jennings, New York City, who was b. 1818, dau. of Nathan Tilestone Jennings and Maria Miller. Their children:

1. Maria Jennings, b. 1838, d. 1840.
2. William Gerard, b. 1840. } Twins.
3. Donald McKenzie, b. 1840, d. 1843. }
4. Anna Margaretta, b. 1843.
5. Anna Catherine, b. 1845.

Figure 1: An Excerpt from Page 419 of *The Ely Ancestry*.

Automating the process of extracting stated facts, inferring implicit facts, and resolving object references is non-trivial. To aid in performing this task, we are developing FROntIER (**F**act **R**ecognizer for **Ont**ologies with **I**nference and **E**ntity **R**esolution), which is a framework for automatically extracting and organizing facts from historical documents into disambiguated entity records. We give an overview of our framework in Section 2. FROntIER makes use of extraction ontologies [ECJ$^+$99, ELL11] to automatically extract stated facts of interest. We describe extraction ontologies in Section 3. FROntIER uses organization rules to organize facts with respect to a target schema. We describe organization rules in Section 4. In Section 5 we present our experimental results over facts found in Figure 1 and add some concluding remarks in Section 6.

2 FROntIER

As stated previously, FROntIER uses extraction ontologies to automatically extract stated facts of interest. Once stated facts have been recognized and properly associated with an extraction ontology, FROntIER disambiguates objects, infers additional facts about these objects, and organizes the objects and facts about these objects with respect to a target ontology. FROntIER's extraction ontologies allow users to model text and layout as it appears in historical documents, while FROntIER's target ontologies model knowledge of interest to be gleaned by historians—facts both directly and indirectly stated. To see the difference, compare the target ontology in Figure 2, which is an ontological view of biographical facts of a person, against the extraction ontology in Figure 3, which models how explicitly stated biographical facts appear in *The Ely Ancestry*. FROntIER uses pattern-based extractors to identify the existence of objects and their interrelationships according to the particular layout in the text document, and uses logic rules to organize extracted facts in a target ontology. FROntIER, for example, extracts the stated "son of" and "dau. of" facts into the *Son-Person* and *Daughter-Person* relationship sets in Figure 3 and then uses the inference rules "if Son, then male" and "if Daughter, then female" to populate the *Person-Gender* relationship set in Figure 2. Inference and organization also include entity resolution, which proceeds based on extracted and inferred facts. The first-mentioned Mary Ely in Figure 1, for example, is the grandmother of the second-mentioned Mary Ely, and therefore cannot be the same Mary Ely.

Our target application is historical documents, which are OCRed pages in PDF format, and input in FROntIER. FROntIER first invokes OntoES, our **Onto**logy **E**xtraction **S**ystem [ECJ$^+$99], which extracts information from pages of text documents and populates an ontol-

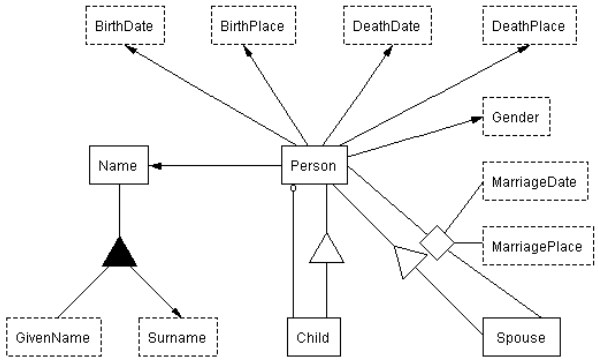

Figure 2: Target Ontology of Desired Biographical Facts.

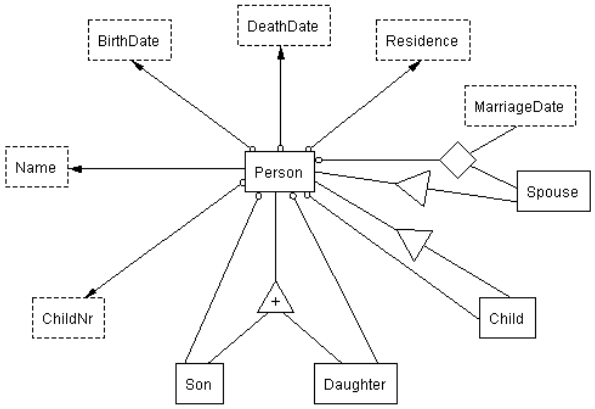

Figure 3: Extraction Ontology of Stated Biographical Facts of a Person in *The Ely Ancestry*.

ogy with recognized objects, object properties, and relationships among objects and object properties. The output of OntoES is an XML document containing these objects and relationships, which is converted into RDF[1] triples (in an OWL[2] ontology) to be processed by the Jena[3] reasoner. We use inference rules, and the Jena reasoner, to transform RDF triples into triples that conform to a target ontology and infer implied facts. The output from the Jena reasoner is converted into an OSM data instance model. We convert the data instances into a comma-separated value file, the input into Duke[4]. FROntIER uses Duke to disambiguate entities, producing *owl:sameAs* relationships to denote that entities are the same.

3 Extraction Ontologies

An extraction ontology is a linguistically grounded conceptual model. Figure 3 is an example of the conceptual model component of an extraction ontology. Figure 4 illustrates some of the linguistic grounding for the conceptualization in Figure 3.

In a conceptual model diagram (e.g. Figures 2 and 3), each box represents an object set, or class of objects. Object sets can either be lexical (represented with dashed lines) or non-lexical (represented with solid lines). Lexical object sets contain strings whereas non-lexical object sets contain surrogates that denote real-world objects. Line segments between object sets denote relationship sets. An unfilled triangle denotes generalization/specialization with the generalization, or object set that represents the hypernym, connected to the apex of the triangle and the specializations, or object set that represents the hyponyms, connected to the base. More details on concepts found in the conceptual model diagrams can be found in [ELL11].

The linguistic component of an extraction ontology consists of four types of instance recognizers—recognizers for lexical object sets, non-lexical object sets, relationship sets, and designated ontology snippets. Instance recognizers are embedded in data frames [Emb80]—abstract data types tied to concepts in an extraction ontology that, in addition to instance recognizers, contain operators that manipulate data values.

Lexical Object Sets:
Name
external representation: \b{FirstName}\s{LastName}\b
...
BirthDate
external representation: \b[1][6-9]\d\d\b
left context: b\.\s
right context: [.,]
...
Non-lexical Object Sets:
Person
object existence rule: {Name}
...
Son
object existence rule: {Person}[.,]?.{0,50}\s[sS]on\b
...
Relationship Sets:
Person-BirthDate
external representation: ^\d{1,3}\.\s{Person},\sb\.\s{BirthDate}[.,]
...
Son-Person
external representation: {Son}[.,]?.{0,50}\s[sS]on\s+of\s.*?\s{Person}
...
Person-MarriageDate-Spouse
external representation: {Person}[.,]?.{0,50};\s*m[.,]\s{MarriageDate}[,]?\s*{Spouse}
...
Ontology Snippets:
ChildRecord
external representation: ^(\d{1,3})\.\s+([A-Z]\w+\s[A-Z]\w+)
(,\sb\.\s([1][6-9]\d\d))?(,\sd\.\s([1][6-9]\d\d))?\.
predicate mappings: *Person-ChildNr*(x,1); *Person-Name*(x,2); *Child*(x);
Person-BirthDate(x,4); *Person-DeathDate*(x,6)

Figure 4: Example Data Frames for Concepts in the Ontology in Figure 3.

Lexical object-set recognizers identify lexical instances in terms of external representations, context, exclusions, and dictionaries. External representations are regular expressions for specifying how instances may appear in text. For example, the external representation for *BirthDate* in Figure 4 represents years between 1600 and 1999. Left context is a regular expression that matches text found immediately before an instance pattern, and likewise, right context is a regular expression that matches text

[1] http://www.w3.org/RDF/
[2] http://www.w3.org/TR/owl2-overview/
[3] http://jena.apache.org
[4] http://code.google.com/p/duke/

found immediately after an instance pattern. Dictionaries are regular expressions where each entry in the dictionary is delimited by an OR ('|'), e.g. for *Name* in Figure 4 "(Abigail|Mary|William|...)" could be part of the *FirstName* dictionary. In general, braces around a name—e.g. {FirstName}—refer to a regular expression defined elsewhere. Non-lexical object-set recognizers identify non-lexical objects through object existence rules. Object existence rules identify text that designates the existence of an object. An example is a person's name. When a name is recognized by OntoES, it generates a *Person* object and associates it with the recognized name. Object existence rules for non-lexical specializations identify roles for objects in its generalization. For example, the object existence rule for *Son* in Figure 4 establishes that a person recognized is a son.

Relationship-set recognizers identify phrases that relate objects. For example, the external representation for *Person-BirthDate* in Figure 4 represents a phrase that relates a person to a birth date. To process the recognizer, OntoES replaces "{Person}" and "{BirthDate}" with strings previously recognized for the *Person* and *BirthDate* object sets. OntoES uses the resulting regular expression to relate Maria Jennings to 1838 and William Gerard to 1840—two of the *Person-BirthDate* relationships that appear in Figure 1.

Ontology-snippet recognizers identify text patterns that provide instances for several object and relationship sets. Data frames for ontology snippets consist of external representations and predicate mappings. External representations are regular expressions with capture groups that map captured instances to ontology predicates—the object and relationship sets in the ontology. Variables for the mappings denote non-lexical objects, and integers denote captured lexical objects. As an example, suppose the ontology-snippet recognizer named *ChildRecord* in Figure 4 is applied to the first child list in Figure 1. The recognizer for *ChildRecord* would identify the pattern "(1). (Mary Ely) (, b. (1836))(, d. (1859))." where the parenthesized expressions represent captured groups numbered left to right by appearance of a left parenthesis. Predicate mappings for *ChildRecord* would generate the following objects and relationships for the ontology in Figure 3 for Mary Ely, the 5th person mentioned in Figure 1:

Person-ChildNr($Person_5$, "1")
Person-Name($Person_5$, "Mary Ely")
Person-BirthDate($Person_5$, "1836")
Person-DeathDate($Person_5$, "1859")
Child($Person_5$)

Plus as implied by referential-integrity and generalization/specialization constraints:

Person($Person_5$)
Name("Mary Ely")
BirthDate("1836")
DeathDate("1859")

4 Organization Rules

FROntIER uses rules to organize facts in with respect to a target ontology (e.g. Figure 2). The kind of organization rules that FROntIER uses include canonicalization, inference, and entity-resolution rules. We briefly explain each kind of rule.

Canonicalization rules homogenize recognized instance values so that they can be manipulated and compared. For example, values for *BirthDate* in Figure 3 such as "1836" and "1832", which are strings, are canonicalized into an internal data type such as *Date*.

Inference rules specify schema mappings between a source ontology and a target ontology. Figure 5 shows several sets of rules in the Jena syntax. Each rule set corresponds to a particular source-to-target transformation. For example, the first rule set in Figure 5 copies the *Person* instances in the ontology in Figure 3 to *Person* instances in Figure 2. This kind of rule performs a direct schema mapping. The second rule set in Figure 5, which also performs a direct schema mapping, copies *BirthDate* instances as well as *Person-BirthDate* instances in the ontology in Figure 3 to *BirthDate* instances and *Person-BirthDate* instances in Figure 2.

```
1 [(?x rdf:type source:Person) -> (?x rdf:type target:Person)]

2 [(?x rdf:type source:BirthDate),(?x source:BirthDateValue ?bv)
  -> (?x rdf:type target:BirthDate),(?x target:BirthDateValue ?bv)]
  [(?x source:Person-BirthDate ?y) -> (?x target:Person-BirthDate ?y)]

3 [(?x source:Person-Name ?n),(?n source: NameValue ?nv), isMale(?nv)
  -> (?x target:Person-Gender ?gender),(?gender rdf:type target:Gender),
     (?gender target:GenderValue 'Male'^^xsd:string)]
```

Figure 5: Example Inference Rules for Organizing Facts with Respect to a Target Ontology.

Our inference rules are constrained to the set of constructs supported by the Jena framework. Conveniently, the Jena framework defines a set of built-in predicates that is extendable. For extending the set of built-ins, the Jena framework allows the implementation of a *builtin* interface, and we implement this interface for each user-defined built-in. For example, the third rule set in Figure 5 infers the gender of a person as male; the user-defined built-in 'isMale' refers to a predefined statistical table to determine whether a name is for a male [Sch12].

FROntIER's entity-resolution rules use facts for entities in populated target ontologies as input and generate *owl:sameAs* relationships as output. FROntIER can use any off-the-shelf or specially developed fact-based entity resolver. We have chosen to use Duke, an off-the-shelf entity resolver.

Duke uses a configuration file to set attribute comparators, parameter values, and a threshold value. We use Jaro-Winkler similarity for comparing name components and exact match comparisons for all other attributes. For parameter values, each attribute has a low value for when two attribute-value pairs do not match and a high value for when they do match. Duke combines parameter values to produce a probability that two entities are the same. If the probability exceeds the threshold value, then the two entities are placed in the same equivalence class. We take the equivalence classes produced by Duke and generate same-as relationships between entities in the equivalence classes.

5 Experimental Results

We calculated accuracy through precision and recall over facts extracted from Figure 1 using an extraction ontology such as the one in Figure 3. The extracted facts were tested against annotated gold standard facts. We calculated accuracy over implied facts in a similar manner using a target ontology such as the one in Figure 2.

Extracted Facts	Precision	Recall	F-measure
Name	1.000	0.941	0.970
BirthDate	0.818	0.818	0.818
DeathDate	0.800	0.800	0.800
MarriageDate	0.500	0.500	0.500
Person-BirthDate	0.889	0.727	0.800
Person-DeathDate	0.750	0.600	0.667
Son-Person	1.000	1.000	1.000
Daughter-Person	1.000	1.000	1.000
Child-Person	1.000	0.857	0.923
Person-Spouse-MarriageDate	1.000	0.500	0.667
Implied Facts	**Precision**	**Recall**	**F-measure**
Person-Name*	0.958	0.902	0.929
Person-BirthDate	0.889	0.727	0.800
Person-DeathDate	0.750	0.600	0.667
Person-Gender	1.000	0.941	0.970
Person-Child	1.000	0.900	0.947
Person-Spouse-MarriageDate-MarriagePlace	1.000	0.900	0.947
Person_A same-as Person_B	1.000	1.000	1.000

*Name is composed of GivenNames and Surnames

Table 1: Accuracy over Facts in Figure 1.

Table 1 shows the accuracy of FROntIER over extracted and implied facts found in Figure 1. The precision and recall values were very high for every type of fact except *Person-DeathDate*. This is due to the difficulty in capturing this type of relationship in a generalized manner, the structure of the records (which placed birth and death date information at the end of the record for Donald McKenzie versus near the beginning for William Gerard Lathrop), the low number of death dates available in the figure, and the difficulty of recognizing names that cross line boundaries (and thus include a hyphen in an unexpected location). Another value of note is the recall for *Person-Spouse-MarriageDate-MarriagePlace* versus *Person-Spouse-MarriageDate* which went up because there were many more inferred marriage relationships than explicitly stated in Figure 1 such as Gerard Lathrop married to Mary Ely and vice versa.

6 Concluding Remarks

Our experimental results show that FROntIER works as a proof-of-concept. More experiments need to be performed before any conclusions may be drawn regarding the time and expertise required to gather all facts of interest from the entire book and at what accuracy level. Our hope is that overall accuracy will prove to be a function of extraction accuracy so that the emphasis will be on improving fact extraction.

References

[BEV02] M.S. Beach, W. Ely, and G.B. Vanderpoel. *The Ely Ancestry*. The Calumet Press, 1902.

[ECJ+99] D.W. Embley, D.M. Campbell, Y.S. Jiang, S.W. Liddle, D.W. Lonsdale, Y.K. Ng, and R.D. Smith. Conceptual-model-based data extraction from multiple-record web pages. *Data & Knowledge Engineering*, 31(3):227–251, 1999.

[ELL11] D.W. Embley, S.W. Liddle, and D.W. Lonsdale. Conceptual modeling foundations for a web of knowledge. In D.W. Embley and B. Thalheim, editors, *Handbook of Conceptual Modeling: Theory, Practice, and Research Challenges*, chapter 15, pages 477–516. Springer, 2011.

[Emb80] D.W. Embley. Programming with data frames for everyday data items. In *Proceedings of the AFIPS National Computer Conference*, pages 301–305, Anaheim, CA, USA, May 1980.

[Sch12] P. Schone. Personal communication, 2012.