A Demonstration of Entity Identity Information Management Applied to Demographic Data in a Referent Tracking System

Cheng Chen[1,*], Josh Hanna[2], John R. Talburt[1], Mathias Brochhausen[2], William R. Hogan[2]
[1] Information Quality Program, University of Arkansas at Little Rock, Little Rock, AR, USA
[2] Division of Biomedical Informatics, University of Arkansas for Medical Sciences, Little Rock, AR, USA

ABSTRACT

Referent Tracking (RT) is an ontology-based approach to tracking individual persons, processes, diseases, prescriptions, etc. Each such individual is assigned a unique identifier, called an Instance Unique Identifier (IUI). Assignment of duplicate IUIs to a single entity is highly problematic in practical applications, as is the assignment of one IUI to two different entities. To address these problems, we applied an entity resolution system to manage the data quality of the RT system. This paper describes a demonstration system that integrates entity resolution techniques and tools into a RT system to solve duplication problems and to track identifiers over time.

1 INTRODUCTION

Referent tracking (RT) is a methodology for capturing high-fidelity representations of (1) particular entities and (2) what various agents know about those entities, over time. To do so, an RT system (RTS) assigns globally-unique identifiers—called Instance Unique Identifiers (IUIs)—to each entity in reality about which it stores information. RT uses a series of templates for unambiguous representation of the relationships of particulars (Ceusters & Manzoor, 2010; Hogan *et al*., 2011a).

Two things, among others, are necessary to running a successful RTS:

- Preventing assignment of multiple IUIs to one individual.
- Preventing assignment of a single IUI to multiple individuals.

Once these errors have occurred, they are hard to detect and correct. Ultimately, these types of errors compromise the quality of data in the RTS and will lead to incorrect information being passed on to its users.

Entity Identity Information Management (EIIM) is one strategy to prevent, detect, and correct these errors in an RTS. EIIM aims at ensuring that each entity in the system has one and only one representation, called an identity (Zhou, 2011). The core of EIIM is the process of determining whether two records in a system are referring to the same real-world object or to different objects. This process is called Entity Resolution (ER) (Talburt, 2011).

Here, we present a system which integrates EIIM with a demographics application built on a RTS. Our system applies the strategies of identity management according to EIIM to manage IUIs to maintain data quality in the RTS.

* Email:cxchen1@ualr.edu

2 BACKGROUND

For our system, three basic ER components are used: identity capture, identity update, and identity resolution. **Identity capture** builds a set of identities by processing a set of pre-existing records. **Identity update** creates new identities and updates existing identities during the processing of new records. **Identity resolution** resolves the input references against pre-existing identities.

In EIIM, Entity Identity Structures (EIS) are built for storing the identity information gathered by identity capture and update processes. In these two processes, EIS are produced based on user-provided logic that is usually implemented as one or more matching rules (Zhou, 2011; Whang et al., 2010). EIS can also be improved by allowing users to assert resolution to bring the external knowledge to force the equivalence (or non-equivalence) of entity references and identity structures.

The Open sYSTem Entity Resolution (OYSTER) (Zhou, 2011) software is an application that implements EIIM. The Entity Resolution and Information Quality group at the University of Arkansas at Little Rock built and maintains it.

3 SYSTEM DESCRIPTION

We designed a web-based system for entering, storing and managing demographics data that incorporates the principles of EIIM. We previously designed a demographics demonstration application based on an RTS (Hogan *et al*., 2011b). Our new system applies OYSTER to this pre-existing application to manage identities of persons. OYSTER is integrated into the RTS by mapping RT templates to corresponding EIS created and maintained by OYSTER, synchronously updating both.

The system was developed using PLAY 2.0.4 (Bort, 2013), a web framework built in Scala. The source code of the system is available at: https://bitbucket.org/cxchen1/rts-demographics-oyster/downloads.

3.1 Preprocessing of the RTS

When the application is started, there are three possible states of the RTS:

1) empty,
2) not empty and with previous OYSTER processing,

3) not empty and without previous OYSTER processing.

The possible processing flows under which an RTS may go, according to its starting state, is shown in Figure 1. Each process builds an EIS file that the system uses for further processing. If the RTS is empty, the system will wait for the first record and run identity capture on it to initialize the EIS file. When the RTS is not empty and has undergone processing by OYSTER previously, the system can easily build the EIS file because the OYSTER ID, a unique 16-character identifier that OYSTER assigns each EIS (Zhou, 2011), is stored in the RTS. The RTS itself will not be changed. If the RTS has not been processed by OYSTER, OYSTER will generate one identity structure for each person in the RTS with associated OYSTER IDs using identity capture.

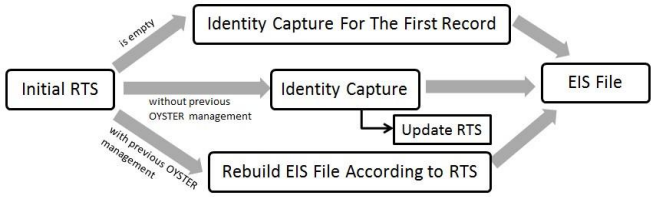

Fig. 1. The possible processing flows based on initial RTS state

3.2 Main Workflow of the System

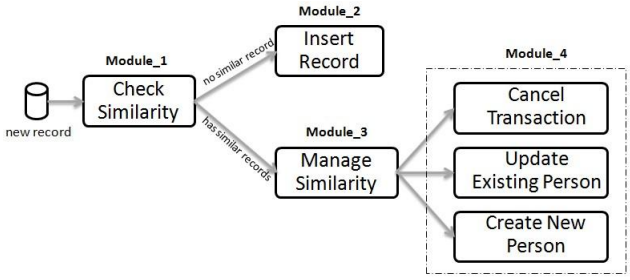

Fig. 2. Overall system workflow.

After the initial preprocessing step, the system is ready for new records. When the user inputs a new set of demographics data using the web application, the main system workflow will begin. There are four main modules in the main workflow (as demonstrated in Figure 2):

Module_1: Identity resolution is first applied to check whether the RTS has any entities OYSTER deems similar to the new entity just entered using the predefined rules.

Module_2: If similar entities are not found, the new data is simply inserted into the RTS. Every additional entity in the RTS triggers a onetime synchronization between the RTS and OYSTER. In the synchronization process, the RTS hands the related IUIs of the new entity and its associated demographic data to OYSTER. After identity update, which updates the EIS file with the new entity, OYSTER transfers the OYSTER ID of the new record to the RTS. RTS will update the entity by adding the OYSTER ID according to Hogan et al (Hogan *et al.*, 2011a).

Module_3: If OYSTER flags other records as similar, then the new record is related to these records in one of three ways: the new record is a duplicate of an existing record, the new record contains updated information for one or more of the similar records, or the new record is completely unrelated. The system will show the user these similar records to allow the user to decide whether these records are really duplicates and thus refer to one person. The system presents the user with three options respectively: cancel the transaction, update the stored entity, or create a new entity.

Module_4: If the user decides to cancel the transaction, the application will go back to "ready status" to wait for a new record. If the user decides to update the existing person's information, the application will perform an identity update, adding new information and invalidating old information as necessary. If the user creates a new person, the entity will be persisted into the RTS and a **structure split assertion**, which allows the user to disassociate the references in one EIS (split) into two or more EIS (Zhou & Talburt, 2010), will be applied so that OYSTER does not associate the new entity with the existing matches in the future.

ACKNOWLEDGEMENTS

This work was funded in part by award UL1TR000039 from the National Center for Advancing Translational Sciences (NCATS). The content of this paper is solely the responsibility of the authors and does not necessarily represent the official views of the NIH.

REFERENCES

Bort,G.(2013).*Play 2.0.* http://www.playframework.com/.

Ceusters, W., Manzoor, S.(2010). *How to track absolutely everything.* In: Obrst L, Janssen T, Ceusters W, editors. Ontologies and Semantic Technolo.

Talburt, J. R.(2011). *Entity Resolution and Information Quality.* San Francisco, CA: Morgan Kaufmann.

Whang, S. & Garcia-Molina, H.(2010) *Entity Resolution with Evolving Rules.* Proceedings of the VLDB Endowment, Vol. 3 Issue 1-2, 1326-1337.

Hogan, W. R., Garimalla, S., Tariq, S. A., Ceusters, W.(2011a). *Representing Local Identifiers in a Referent-Tracking System.* ICBO: International Conference on Biomedical Ontology, Buffalo, NY, USA. p.252-254.

Hogan, W. R., Garimalla, S., Tariq, S. A. (2011b) *Representing the Reality Underlying Demographic Data.* ICBO: International Conference on Biomedical Ontology. Buffalo, NY, USA. p.147-152.

Zhou,Y.(2011). *Entity Identity Information Management (EIIM).* Proceedings of the 16th International Conference on Information Quality, p.327-341.

Zhou, Y., and Talburt, J.R.(2011). *The Role of Asserted Resolution in Entity Identity Management.* Proceedings: The 2011 International Conference on Information and Knowledge Engineering (IKE'11).pp.291---296.