ID Match Engine

ID Match Engine

Overview

A web service, the function of which is to return identity matches for a given identity search input. The search response will be based on a set of match rules as configured by the tool administrator.The web service supports many popular and accepted Identity/Entity Matching Algorithms (both Phonetic and Edit Distance) such as Soundex, NYSIIS, DaitchMakotoff, JaroWinkler, LevenShtein. The implementation of these algorithms is based on OYSTER project from University of Arkansas, the details of which are provided in the next section.

ID Match Core - Oyster

The ID Match core reuses Libraries from the OYSTER project. The OYSTER Open Source Project is sponsored by the Center for Advanced Research in Entity Resolution and Information Quality (ERIQ) at the University of Arkansas at Little Rock. It is intended to provide an entity resolution system that includes functionality for entity identity information management (EIIM). For information, please goto the References section below.

In addition to the OYSTER, we evaluated other open source projects in this domain, particularly FRIL(Emory University) and OpenEMPI(Master Patient Index). OYSTER was selected to be the ideal candidate to reuse as it provided for the most current and very well documented code base. Besides of all the three projects selected as finalists, OYSTER is the only project that is quite active and still releasing updates.

Though the said matching algorithms are pretty standard and the implementations for which might not change, we look forward to work with UA at LittleRock, for any help in tuning the implementations for performance gains where needed.

ID Match APIs

The ID Match Engine tool is designed to provide for a Rest like service with JSON bindings. The implementation of which is currently under testing. The API is not stable and is subject to change based on developer testing and community feedback.

For more details on the JSON request/response signature, Please refer to https://spaces.at.internet2.edu/display/cifer/SOR-Registry+Strawman+ID+Match+API

Note: While the API implementation is worked on, the tool provides for a simple web interface to test out its capabilities, such as configuring match rules, providing a search input and reviewing the results.

ID Match Configuration Management

The main and most significant configuration task involves creating a set of Matching Rules. It is here the administrator will create a Rule that specifies the attribute to match, the algorithm to use for the matching and the score to be given when such a condition is met. The administrator shall also provide for a set of scores that will be used to classify a match as either exact match or a match for manual reconciliation.

ID Match Administrative Tools/UI

The current revision of the application has web based Administrative UI, which is slated to deprecated in favor of file based configuration.

Code

ID Match Github Repo

References

OYSTER:http://sourceforge.net/p/oysterer/home/Home/

FRIL:http://fril.sourceforge.net/

OpenEMPI:http://sourceforge.net/projects/openempi/http://openempi.kenai.com/

Next steps

1. Needs extensive performance testing and benchmarking.
2. Bug fixes around Match Rule configuration. Currently can create rules with scores exceeding 100.
3. Change the configuration storage from backend to file based. Currently configuration is stored in backend tables. Will be moved to config file.
4. Change configuration administration from web UI to config file. Web UI will be deprecated in favor of config files.
5. Provide for configurable Schema. Current version supports a limited set of attributes hard coded into the app.
6. Provide for interface to seed the backend identity registry.
7. Edit Distance algorithms are hard coded to use 2 as the distance length, this will be changed to configurable setting.

I'm Interested, How do I get involved?

1. Read all the stuff here, watch the presentations in the evaluation section

2. Send comments, questions, ideas to the CIFER Registries Workstream: osidm4he-registries@internet2.edu
3. Join the community mailing lists for Registry projects you might want to participate in
4. Go to the github rep and follow the readme instructions to try out the development ID Match Engine
5. Add comments here
6. If you have a specific project you are working on, consider joining and participating in the CIFER Registry workstream group