GEM: The GAAIN Entity Mapper

Naveen Ashish[1(✉)], Peehoo Dewan[1], Jose-Luis Ambite[2], and Arthur W. Toga[1]

[1] Laboratory of NeuroImaging, Keck School of Medicine of USC, USC Stevens Neuroimaging and Informatics Institute, University of Southern California, 2001 N Soto Street, Los Angeles, USA
{nashish,pdewan,toga}@loni.usc.edu
[2] Information Sciences Institute, University of Southern California, 4676 Admiralty Way, Marina del Rey, Los Angeles, CA 90292, USA
ambite@isi.edu

Abstract. We present a software system solution that significantly simplifies data sharing of medical data. This system, called GEM (for the GAAIN Entity Mapper), harmonizes medical data. Harmonization is the process of unifying information across multiple disparate datasets needed to share and aggregate medical data. Specifically, our system automates the task of finding corresponding elements across different independently created (medical) datasets of related data. We present our overall approach, detailed technical architecture, and experimental evaluations demonstrating the effectiveness of our approach.

1 Introduction

This paper describes a software solution for medical data *harmonization.* Our work is in the context of the "GAAIN" project in the domain of Alzheimer's disease data. However, this solution is applicable to any medical and clinical data harmonization in general. GAAIN stands for the Global Alzheimer's Association Interactive Network[1], a data sharing federated network of Alzheimer's disease datasets from around the globe. The aim of GAAIN is to create a network of Alzheimer's disease data, researchers, analytical tools and computational resources to better our understanding of this disease. A key capability of this network is also to provide investigators with access to harmonized data across multiple, independently created Alzheimer's datasets.

Our primary interest is in medical data sharing and specifically data that is harmonized in the process of sharing. Harmonized data from multiple data providers has been curated to a unified representation after reconciling the different formats, representation, and terminology from which it was derived [7, 16]. The process of data harmonization can be resource intensive and time consuming and our work is a software solution to significantly automate that process. Data harmonization is fundamentally about data alignment - which is to establish correspondence of related or identical data elements across different datasets. Consider the very simple example of a data element capturing the gender of a subject that is defined as 'SEX' in one dataset, 'GENDER' in another

[1] http://www.gaain.org.

N. Ashish and J.-L. Ambite (Eds.): DILS 2015, LNBI 9162, pp. 13–27, 2015.
DOI: 10.1007/978-3-319-21843-4_2

and 'M/F' in yet another. When harmonizing data, a unified element is needed to capture this gender concept and to link (align) the individual elements in different datasets with this unified element.

The data mapping problem can be solved in two ways. We could map elements across two datasets, for instance match the element 'GENDER' from one data source (DATA SOURCE 1) to the element 'SEX' in a second source (DATA SOURCE 2). We could also map elements from one dataset to elements from a *common data model*. A common data model [7] is a uniform representation which all data sources or providers in a data sharing network agree to adopt. The fundamental mapping task is the same in both. Also, the task of data alignment is inevitable regardless of the data sharing model one employs. In a *centralized data sharing* model [15], where we create a single unified store of data from multiple data sources, the data from any data source must be mapped and transformed to the unified representation of the central repository. In *federated* or *mediated* approaches to data sharing [7] individual data sources (such as databases) have to be mapped to a "global" unified model through mapping rules [1]. The common data model approach, which is also the GAAIN approach, also requires us to map and transform every dataset to the (GAAIN) common data model. This kind of data alignment or mapping can be a multi-month effort *per dataset* in medical and clinical data integration case studies [1]. A single dataset typically has thousands of distinct data elements of which a large subset needs to be accurately mapped. On the other hand it is well acknowledged that data sharing and integration processes need to be simplified and made less resource intensive for data sharing in the medical and clinical domains [1, 7]) as well as the more general enterprise information integration domain [10]. The GEM system is built to achieve this by providing automated assistance to developers for such data alignment or mapping.

The GEM data mapping approach is centered on exploiting the information in the data documentation, typically in the form of *data dictionaries* associated with the data. The importance of data dictionary documentation, and for Alzheimer's data in particular, has been articulated in (Morris et al., 2006). These data dictionaries contain detailed descriptive information and metadata about each data element in the dataset. The rest of this paper is organized as follows. In the next section (Sect. 2) we review the work and available industrial or open-source software tools that are related to data mapping. This is followed by a detailed description of the GEM system. In Sect. 4 we present experimental results evaluating the efficacy of the GEM system and also a detailed comparison with related data mapping systems. Finally we propose further work and provide a conclusion.

2 Related Technologies

Data mapping is often done manually based on data dictionaries, on any other information such as database design diagrams [9], and in consultation with the original dataset creators and/or administrators. Data mapping is well understood (Halevy et al., 2005) and there are a number of software tools that have been developed in the past years that relate to it. We first examine existing software tools to (1) determine their applicability to our domain, (2) understand what functions are still needed in the GEM system.

Existing tools can be categorized as metadata visualization tools, Extract-Transform-Load (ETL), and schema-mapping tools. *Metadata visualization tools* are those that create a visual representation of the design of a database by examining the database itself. For instance SchemaSpy[2] provides functionality of "reverse engineering" to create a graphical representation of metadata, such as an "ER" (Entity-Relationship) diagram [9] from the database metadata. Altova[3] is a tool for analyzing and managing relationships among data in data files in XML. These tools are relevant to our task as they can be employed to examine the data and/or metadata of a new dataset that we have to map. *Extract-Transform-Load* (*ETL)* tools provide support for data schema mapping. However the mappings are not automated and have to be created by hand using a graphical user interface (GUI). Tools in this category include Talend[4], Informatica[5] and Clio (Haas et al., 2000). The category most relevant to our data mapping problem is *Schema-Mapping* which provides automated mapping of data elements from two different database or ontology schemas. These tools take as input the data definition language or "DDL" [9] associated with a dataset (database) and are able to match elements across two database schemas based on the DDL information. Prominent examples in this category include the Harmony schema-mapping tool[6] from the Open Information Integration or OpenII initiative and Coma++ (Rahm et al., 2012). There are also schema-mapping tools that are based on "learning-from-examples" i.e., the system is trained to recognize data element mappings from a tagged corpus of element matches (from the domain of interest). LSD [8] is an example in this category. Another tool is KARMA[7] which actually has more of an ontology alignment focus as opposed to data (element) mapping. Finally, PhenoExplorer [8], is an online tool that allows researchers to identify research studies of interest. Specifically, a researcher can search for studies along a set of dimensions, including race/ethnicity, sex, study design, type of genetic data, genotype platform, and diseases studied and the system determines the relevance of a study by mapping data elements in a study to dimensions specified by a researcher.

Our work was motivated by the observation that the rich metadata available in data dictionaries of medical datasets can be leveraged towards a significantly more automated approach to schema-mapping than could be done with existing tools. The next section describes the details of our approach.

3 Methods

This section describes our approach and the technical details of the GEM system. We begin with enumerating the particular data characteristics of Alzheimer's disease and

[2] http://schemaspy.sourceforge.net.
[3] http://www.altova.com.
[4] http://www.talend.com.
[5] http://www.informatica.com.
[6] http://openii.sourceforge.net.
[7] http://www.isi.edu/integration/karma/.

medical data schemas as they bear upon the data mapping approach. We also describe the metadata detail that is typically present in medical data dictionaries that can be accommodated. We then present the GEM architecture and description of the algorithms.

3.1 Medical Data Characteristics

Medical data and associated data schemas have the following characteristics that are relevant to the schema mapping problem:

(i) **Availability of Metadata but not Data**. Overall, data providers may be more willing to make metadata (dictionaries) available during harmonization but the not the actual data. Alzheimer's and other medical research data are highly sensitive and data providers are typically willing to share their metadata (such as data dictionaries) but actual access to data may be restricted. In fact many data sharing and exploration networks help users to locate relevant data and cohorts but actual data must be obtained directly from data providers (Mandel et al., 2012). The data harmonization and thus the data mapping process must work with the metadata (only), and not assume the availability of actual data. This is an important distinction as some schema mapping tools, such as Coma ++, expect the availability of actual data (as well) to generate mappings.

(ii) **Element Names and Element Descriptions.** Data elements often have cryptic names in medical datasets. An example is 'TR1S1' which is ill defined and difficult to infer. The element names can also be *composite*. Essentially, a data element may be one of an entire family of elements. For instance an element named 'MOMDEMYR1' has 3 sub-elements in the name which are MOM (for mother), DEM (for dementia) and YR1 for year 1. Element names thus are of limited utility in determining element mappings in this domain. On the other hand the element descriptions are often rather clear and detailed for each data element and we leverage that for mapping.

(iii) **Presence of Special "Ubiquitous" Data Elements.** There are elements such as the subject identifier, date and timestamp fields, or subject visit number fields that are present in *every* database table in a database. Such elements must be pre-identified and filtered before matching, as they are not candidate matches for other "regular" data elements we seek to match.

3.2 Element Metadata

Relative to other domains such as enterprise data, medical metadata is richer in terms of element descriptions and also accompanying information about the element data type and constraints on values. The detailed metadata that can be extracted or derived from the dictionary information is as follows:

(i) Element Description. We usually have a text description of what the element fundamentally is. In the example in Fig. 1 this is the text under the 'Short

Descriptor' and 'UDS Question sections' (UDS refers to the Uniform Data Set of clinical and cognitive variables in Alzheimer's disease data). The description is usually comprehensive and verbose to the extent required, as opposed to data schemas in other domains where the element (database column) descriptive information (the 'COMMENT' in a DDL) is simply absent or is typically terse.

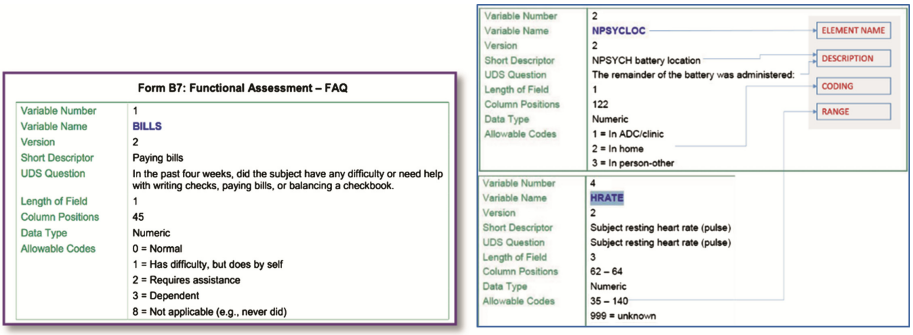


Fig. 1. Element metadata from data dictionary

- Data value constraints. For a majority of data elements, the metadata also contains constraints on the actual values they can take. This information is of two types:
- Coding legend information. The coding legend provided under 'Allowable Codes' tells us the interpretation of various codes, which is the set of possible values that element can take. We can also derive the number of distinct possible values for that element, which is 5 values (0,1,2,3,8) in this example.

(ii) The Range of Values. For many numerical elements, the metadata provides the explicit range of allowable values, for instance the range 0–30 for 'MMSE' scores, etc. MMSE stands for the Mini-Mental State Examination and is commonly used to measure cognitive impairment (Escobar et al., 1986).

(iii) The Element *Category*. Elements can be divided into a few distinct categories based on the kind of values they can take. For instance the element may take one of small set of prefixed codes as values (as in Fig. 1), or take a numerical value such as the (actual) heart rate, etc., This category can be derived from the metadata and is described in more detail below.

All of the above element information is utilized during data mappings, as we describe.

3.3 System

Before describing the system we clarify some terminology and definitions. A dataset is a *source* of data. For instance a dataset provided by ADNI would be a source. A *data dictionary* is the document associated with a dataset, which defines the terms used in

the dataset. A *data element* is an individual 'atomic' unit of information in a dataset, for instance a field or a column in a table in a database or in a spreadsheet. The documentation for each data element in a data dictionary is called *element metadata* or *element information.* A *mapping* or element mapping is a one-to-one relationship across two data elements, coming from different sources. Mappings are created across two distinct sources. The element that we seek to match is called the *query element*. The source we must find matches *from* is called the *target source* and the source of the query element is called the *query source*. Note that a common data model may also be treated as a target source.

The key task of the GEM system is to find element mappings with a "match" operation. "match" is an operation which takes as input (i) a query element, (ii) a target source, and (ii) a matching threshold. It returns a set of elements, from the target source, that match the source element and with a match confidence score associated with each matched element.

Figure 2 illustrates the high level steps of the system. The first step is the *metadata ingestion* step where we start from data dictionaries, extract and synthesize detailed metadata from the data dictionaries for each data element, and store the synthesized metadata in a database. This database is called the *metadata database*. The second step is the *element matching* step where matching algorithms find matches for data elements based on the information in the metadata database.

Fig. 2. System phases

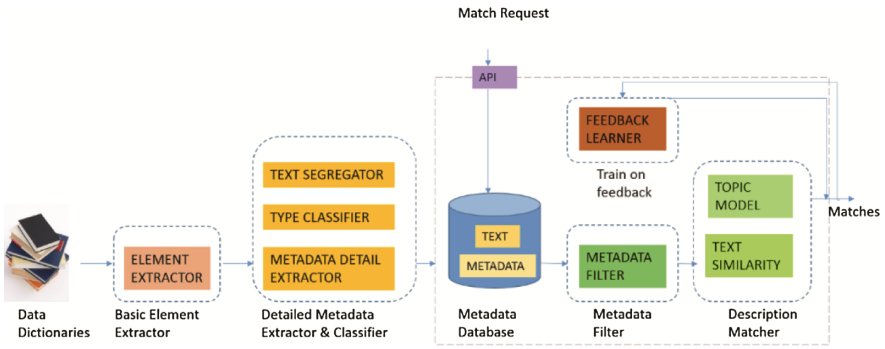


Fig. 3. System architecture

Figure 3 illustrates the architecture and key modules in more detail.

3.3.1 Metadata Ingestion

This part of the pipeline is comprised of two modules. One is for basic individual element metadata extraction from the data dictionary. The other synthesizes detailed metadata per data element.

Basic Element Extractor. The element extractor identifies the description and metadata per data element. In many cases the data dictionary is available in a structured format, such as a spreadsheet, with various components such as the data element name, any (text) descriptions(s) of the field, and other information such the allowable values for the data element etc., clearly delineated. If structured metadata is available this step is not required, however there are instances when data dictionaries are available only as Word or PDF documents. We have developed element extractors for Word and PDF formats to work with these semi-structured documents and extract the per element metadata.

Detailed Metadata Synthesizer. The detailed metadata synthesizer has three components. (1) The first segregates the various important portions of the element overall metadata. (2) The second classifies the data element into a distinct category. (3) The final component extracts specific data constraints that may have been specified for the data element. We describe these.

Segregator: As illustrated in Fig. 1, we model the element information comprised of 4 segments, namely:

(i) The element or field **name**.
(ii) The text **description** of the element, which is the "Short Descriptor" as well as "UDS Question" in the above example data dictionary.
(iii) The value **coding legend**, for applicable elements.
(iv) The value **numerical range** (if any) for a numerical element.

For many data dictionaries segmentation is already complete if the data dictionary itself is structured with various segments in segregated fields. For other formats, such as the example in Fig. 1 (which is a PDF document) we use simple semi-structured data extraction techniques exploiting the labels for the various segments.

Category classifier: The type information of an element (Data Type') illustrated in Fig. 2 is usually provided. We categorize a data element based on the kinds of values it can take. Data elements fall into one of the below categories:

(i) **Coded** elements i.e., where the data values are specific codes for a small finite set of values. Coded elements can be:
    a. **Binary** coded elements i.e., elements that take a Yes/No value
    b. **Other** coded elements
(ii) **Numerical** elements that take a non-coded, actual numerical value. Examples are elements such blood pressure or heart rate.
(iii) **Text** elements that take an actual text value.

We developed an element category classifier that is driven by heuristics as follows:

- Coded elements can be identified by the presence of a coding legend in the element metadata.
- Coded elements can further be classified as Binary Coded elements if they contain legend values such as Yes/No, Present/Absent, 0/1, Normal/Abnormal etc.,
- Numerical elements have a (data) type for numbers (such as integer, float etc.). Also a range is usually specified for numerical elements.
- Text elements have a data type for text strings.
- Special elements
- Elements for date or timestamps are identified by appropriate regular expression patterns
- (Subject) identifier elements are identified by the element name, usually having indicators such as 'ID' in the name.

Metadata Detail Extractor: Here we extract and synthesize the metadata details, specifically, (i) The element cardinality (number of distinct possible data values) from the coding legend, and (ii) The range (minimum and maximum permissible values) for numerical values. This extraction and derivation (for cardinality) is performed using simple regular expression based extraction patterns, and label information.

3.3.2 Metadata Database

The metadata database is a uniform, detailed repository of the extracted metadata. This metadata database powers the various matching algorithms in the matching phase.

3.3.3 Matching

The matching step has two sub-steps as follows:

(1) A *candidate elimination* or *blocking* sub-step, where for a given data element we eliminate *incompatible* candidate elements from consideration. The incompatibility is determined using some metadata details. This step is analogous to *blocking* in record linkage where incompatible or improbable candidates are eliminated in a filtering step (Minton et al., 2005).
(2) A *similarity matching* sub-step, where we determine similarity among compatible candidate elements (to the original element we are seeking a match for) based on the element description.

Incompatible Candidate Blocking. Incompatible candidates can be identified in different ways. The first, applicable to all data elements, is if the original element and the candidate match element have incongruent (different) categories. So essentially all candidates with element category other than that of the original element are incompatible. Candidates can then further be eliminated based on the other metadata constraints, specifically cardinality or range. The cardinality of an element applies to elements where the data values take one of a fixed and finite set of values, typically the set of values is small. The cardinality of the element is then the number of possible such data values it can

take. The cardinalities of two matching elements need to be "close" but not necessarily exactly equal. For instance one data source may have a GENDER element with cardinality of 3 (taking values 'M', 'F', or 'U' for unknown) whereas another source may have a corresponding (gender) element with cardinality of 4 (say 1 each for male and female, 1 for unknown, and 1 for error). For a given element with cardinality O we assume that the cardinalities of any corresponding elements are distributed *normally* with O as the mean and a standard deviation of 1. For a candidate element, with cardinality O', we compute the probability that O' belongs to the normal distribution with $\mu = O$ and $\sigma = 1$. Candidates with this probability below a certain threshold are eliminated.

Candidates in the numerical category can be eliminated based on a range of values. Certain elements have a strict fixed range, by definition, in any dataset. For instance the MMSE score element by definition takes values 0–30 (only). On the other hand an element for heart rate may have a range specified as 35–140 in one dataset and 30–150 in another, both being "reasonable" range bounds for the values. We employ a *range match score* (RMS) that is defined as follows:

$$RMS\,(e1, e2) = \frac{|\min\,(U1, U2) - \max\,(L1, L2)|}{\max\,(U1 - L1, U2 - L2)}$$

This RMS score is measure of the overlap of the range of values across two elements. Candidates with an RMS score below a certain threshold are eliminated.

Similarity Matching. After candidate elimination based on metadata constraints we compute an element similarity match based on the similarity of the element text descriptions. We mentioned that the element (text) description is relatively more comprehensive and verbose in medical data dictionaries and this is the reason we have explored and utilized more sophisticated approaches to determine element description similarity across two elements. Our approach employs *topic modeling* on the element descriptions. Topic modeling (Blei 2012) is an unsupervised machine learning approach, which is used for discovering the abstract "topics" that occur in a collection of documents (data dictionaries). The underlying hypothesis is that a document is a mixture of various topics and that each word in the document is attributable to one of the document's topics. We formally define a topic to be a probability distribution over the unique words in the collection. Topic modeling is a generative statistical modeling technique which defines a joint probability over both observed and hidden random variables. This joint probability is used to calculate the conditional distribution of the hidden variables given the observed variables. In our case, the documents in the collection are the observed variables whereas the topic structure which includes both the topic distribution per document and the word distribution per topic is latent or hidden.

In our approach, each column from the source is considered as a document, with the column name as the document name and the column description as the content of the document. After formatting our input in this way and generating a topic model, we receive a document distribution probability matrix where each row represents a document, each column represents a topic, and each particular document topic cell contains the probability that the particular document belongs to that particular topic. Thus we have for each document i.e., element description, a probability distribution over the set

of topics. The similarity between two element descriptions is the cosine similarity or dot product [18] of the topic probability distribution vectors associated with the two element descriptions. The description similarity (DS) is defined as:

$$DS(e1, e2) = TPV(e1.description) \cdot TPV(e2.description)$$

where TPV = Topic Probability Vector (associated with an element description).

4 Results

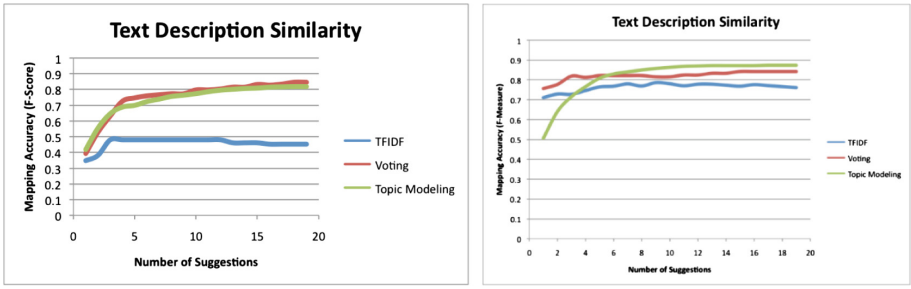
We conducted a series of experimental evaluations with the GEM system which are centered on evaluating the mapping accuracy of GEM with various data schema pairs. Specifically, we determined (i) The optimal configuration for the GEM system that results in high mapping accuracy, (ii) The actual data mapping accuracy that can be achieved by GEM for various GAAIN dataset pairs, and (iii) Comparison of mapping accuracy of GEM with that of other schema-mapping systems.

Experimental Setup. We used six of the data sources of Alzheimer's disease data that we have in GAAIN namely (1) the Alzheimer's Disease Neuroimaging Initiative (ADNI) [17], (2) the National Alzheimer's Coordinating Center database (NACC) [3], (3) the Dominantly Inherited Alzheimer Network database (DIAN) [13], (4) the Integrated Neurogenerative Disease Database (INDD) [21], (5) the Layton Aging and Alzheimer's Disease Center database [20] and (6) the Canadian Longitudinal Study of Aging (CLSA)[8]. The original data provider provided the data dictionaries for each source. We conducted multiple data mappings using GEM, for various pairs of the six datasets as well from the datasets (one at a time) to the GAAIN common model. We also conducted data mappings for some of these dataset pairs using the Harmony system, for comparison. We manually created truth sets of data mappings across these dataset pairs, which are used as the gold standard against which GEM generated mappings are evaluated.

Mapping Accuracy Evaluations. The GEM system provides multiple alternatives as suggested matches for a given data element. The (maximum) number of alternatives provided is configurable. We present results showing data mapping accuracy as a function of the number of alternatives for a set of evaluations below.

Topic modeling vs TFIDF. The first set of evaluations is to determine the effectiveness of topic modeling based text description by evaluating the impact of the text description match algorithm on the mapping accuracy. In addition to topic modeling based text match we also employed a TF-IDF Cosine similarity (Tata and Patel, 2007) algorithm for matching text descriptions. The mapping accuracies for various schema pairs are shown in Fig. 4.

[8] http://www.cihr-irsc.gc.ca.



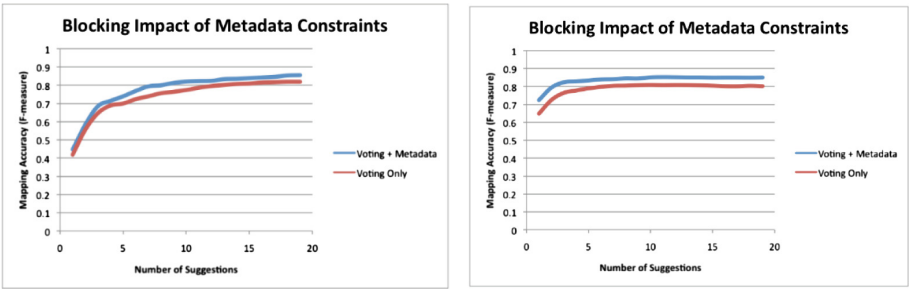
(a) LAADC to ADNI (b) NACC to ADNI

Fig. 4. Text description similarity algorithm impact

Our results with various pairs of schemas, of which the three pairs illustrated in Fig. 4 are a subset, show that in most cases the mapping accuracy achieved with topic modeling based text description matching is superior to that achieved with TFIDF based text matching. This is however not the case universally as in the INDD to ADNI mapping (not illustrated) TFIDF based mapping outperformed that based on topic modeling. Our observation is that topic modeling based text matching works better when the two sources (being matched) have comprehensive data dictionaries with verbose text descriptions for data elements. On the other hand TFIDF appears to work better when one or both data sources have dictionaries with brief or succinct element text descriptions. While not obvious, this result is not surprising given that the underlying topic model generation algorithm, Latent Dirichlet Allocation (LDA), works by finding cohesive themes in large collections of unstructured data (Blei 2012). More elaborate element text descriptions provide a better basis for this algorithm to discover themes in the corpus of all descriptions. In Fig. 4 also show results for an approach that combines TFIDF and topic modeling text match. We use a voting algorithm that considers, for a specific matching instance, either one of topic modeling or TFIDF for determining the text similarity based on which of the two text matching approaches has a higher text match similarity score. The text match similarity score is in the range 0–1 for both approaches. A more principled way to address this however would be to assess the probabilistic confidence that a pair of elements match, given the match similarity scores from both TFIDF and topic modeling approaches. We propose to add this as part of the larger effort of incorporating machine-learning techniques into the system that we discuss in the Conclusions section.

Impact of Blocking Based on Metadata Constraints. Figure 5 illustrates the impact of employing metadata data constraint based filtering or blocking on mapping where we evaluate mapping accuracy with and without the metadata based blocking step.

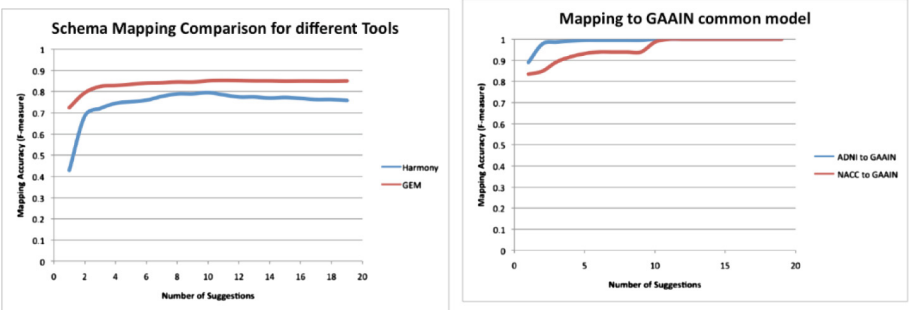
We see that using metadata constraint based blocking indeed provides an improvement in mapping accuracy. The improvement is about 5 % on average and as high as 10 % in some cases as evaluated by mapping across various schema pairs.



(a) LAADC to ADNI (b) NACC to ADNI

Fig. 5. Impact of metadata Constraint Based Blocking

Comparison with Other Systems. We also compared our system with related systems to the extent we could, given limitations of other systems. Our aim was to compare the mapping accuracy of various schema pairs provided to GEM as well as to systems with identical functionality namely Harmony and Coma++. For Harmony, we could complete this comparison for only one of the schema pairs as the system could not work with other schema pairs, given its limitations in terms of the total number of database tables and columns it can reason with. That comparison, NACC to ADNI, is provided in Fig. 6(a) where GEM was significantly superior (around 12–15 % better) than Harmony in mapping this dataset pair. With Coma++, the mapping accuracies for all dataset pairs were less than an F-Measure of 0.3 and we do not report these results. Coma ++ is not designed to consider element text descriptions in schema mapping and the focus is more on matching ontology and XML schemas based on structural information (Bosch et al., 2011).



(a) Comparison with Harmony (b) Common Model Mapping

Fig. 6. Comparison, and mapping to GAAIN common model

Mapping to GAAIN Common Model. Finally, we evaluated the mapping accuracy of GEM to the current GAAIN common data model. The GAAIN common data model currently comprises of 24 data elements of key subject data elements that include demographic elements such as age and gender and also select patient assessments and scores. We represented the common model as (just) another data schema. The results of the mapping from ADNI to GAAIN and NACC to GAAIN are shown in Fig. 6(b).

4.1 Conclusions from Results

The experimental results provide several important conclusions regarding the performance and the configuration of GEM. The GEM system provides high mapping accuracy, in the range of 85 % or above F-Measure for GAAIN datasets and the common model, and for reasonable result window sizes of 6 to 8 result alternatives. The system performs better than existing systems such as Harmony, in terms of both scalability in handling large data schemas as well as mapping accuracy. From a system configuration perspective we can conclude that it is indeed beneficial to determine element text description similarity using a sophisticated topic modeling based approach. This generally results in higher schema mapping accuracies, compared to using existing text similarity techniques. Further, it is advantageous to train the topic model used for text matching, on element text descriptions from a large number of data sources. Finally, metadata constraint based blocking is beneficial in achieving higher accuracy of mapping.

5 Conclusion

We described and evaluated the GEM system in this paper. Compared to existing schema mapping approaches, the GEM system is better optimized for medical data mapping such as in Alzheimer's disease research. Our experimental evaluations demonstrate significant mapping accuracy improvements that have been obtained with our approach, particularly by leveraging the detailed information synthesized from data dictionaries.

Currently we are integrating the GEM system with the overall GAAIN data transformation platform so that developers can operationally use the mapping capabilities to integrate new datasets. We are also enhancing the system with machine-learning based classification for schema mapping. This will enable us to systematically combine various match indicators such as text similarity using multiple approaches such as topic modeling and TFIDF cosine similarity, and also features based on data element name similarity. We are also developing an active learning capability (Rubens, Kaplan and Sugiyama, 2011) where developers can vet or correct GEM system mappings and the system is able to learn and improve from such feedback.

References

1. Ashish, N., Ambite, J.L., Muslea, M., Turner, J.A.: Neuroscience data integration through mediation: an (F)BIRN case study. Front. Neuroinform. 4:118 (2010). doi: 10.3389/fninf.2010.00118. PUBMED PMID: 21228907 PMCID: PMC3017358

2. Aumueller, D., Do, H.H., Massmann, S., Rahm, E.: Schema and ontology matching with COMA++. In: Proceedings of the 2005 ACM SIGMOD International Conference on Management of Data, pp. 906–908. ACM, June 2005
3. Beekly, D.L., Ramos, E.M., Lee, W.W., et al.: The National Alzheimer's Coordinating Center (NACC) database: the uniform data set. Alzheimer Dis. Assoc. Disord. **21**, 249–258 (2007)
4. Blei, D.M.: Probabilistic topic models. Commun. ACM **55**(4), 77–84 (2012). doi: 10.1145/2133806.2133826. http://doi.acm.org/10.1145/2133806.2133826
5. Bosch, T., Mathiak, B.: Generic multilevel approach designing domain ontologies based on XML schemas. In: Workshop Ontologies Come of Age in the Semantic Web, pp. 1–12 (2011)
6. Do, H.H., Melnik, S., Rahm, E.: Comparison of schema matching evaluations. In: Chaudhri, A.B., Jeckle, M., Rahm, E., Unland, R. (eds.) Web, Web-Services, and Database Systems 2002. LNCS, vol. 2593, pp. 221–237. Springer, Heidelberg (2003)
7. Doan, A., Halevy, A., Ives, Z.: Principles of Data Integration. Elsevier, Amsterdam (2012)
8. Doan, A., Domingos, P., Halevy, A.Y.: Reconciling schemas of disparate data sources: a machine-learning approach. In: ACM Sigmod Record, vol. 30, no. 2, pp. 509–520. ACM, May 2001
9. Garcia-Molina, H.: Database Systems: The Complete Book. Pearson Education, India (2008)
10. Halevy, A.Y., Ashish, N., Bitton, D., Carey, M., Draper, D., Pollock, J., Sikka, V.: Enterprise information integration: successes, challenges and controversies. In: Proceedings of the 2005 ACM SIGMOD International Conference on Management of Data, pp. 778–787. ACM, June, 2005
11. Karlawish, J., Siderowf, A., Hurtig, H., Elman, L., McCluskey, L., Van Deerlin, V., Lee, V.M., Trojanowski, J.Q.: Building an integrated neurodegenerative disease database at an academic health center. Alzheimer's Dement. **7**, e84–e93 (2011). doi: 10.1016/j.jalz.2010.08
12. Mandel, A.J., Kamerick, M., Berman, D., Dahm, L.: University of California Research eXchange (UCReX): a federated cohort discovery system. In: 2012 IEEE International Conference on Healthcare Informatics, Imaging and Systems Biology, p. 146 (2012)
13. Morris, J.C., Weintraub, S., Chui, H.C., Cummings, J., DeCarli, C., Ferris, S., Foster, N.L., Galasko, D., Graff-Radford, N., Peskind, E.R., Beekly, D., Ramos, E.M., Kukull, W.A.: The Uniform Data Set (UDS): clinical and cognitive variables and descriptive data from Alzheimer Disease Centers. Alzheimer Dis. Assoc. Disord. **20**(4), 210–216 (2006)
14. Morris, J.C., et al.: Developing an international network for Alzheimer's research: the Dominantly Inherited Alzheimer Network. Clin. Invest. (Lond) **2**(10), 975–984 (2012). PMCID: PMC3489185
15. NDAR: National Database of Autism Research (2014). Web: http://ndar.nih.gov
16. Ohmann, C., Kuchinke, W.: Future developments of medical informatics from the viewpoint of networked clinical research. Methods Inf. Med. **48**(1), 45–54 (2009)
17. Shen, L., Thompson, P.M., Potkin, S.G., Bertram, L., Farrer, L.A., Foroud, T.M., Green, R.C., Hu, X., Huentelman, M.J., Kim, S., Kauwe, J.S., Li, Q., Liu, E., Macciardi, F., Moore, J.H., Munsie, L., Nho, K., Ramanan, V.K., Risacher, S.L., Stone, D.J., Swaminathan, S., Toga, A.W., Weiner, M.W., Saykin, A.J.: Generic analysis of quantitative phenotypes in AD and MCI: imaging, cognition and biomarkers. Brain Imaging Behav. **8**(2), 183–207 (2014)
18. Sidorov, G., Gelbukh, A., Gómez-Adorno, H., Pinto, D.: Soft similarity and soft cosine measure: similarity of features in vector space model. Computación y Sistemas **18**(3), 491–504 (2014). doi: 10.13053/CyS-18-3-2043. Accessed 7 October 2014
19. Tata, S., Patel, J.: Estimating the selectivity of tf-idf based cosine similarity predicates. SIGMOD Rec. **36**(2), 75–80 (2007)

20. Wu, X., Li, J., Ayutyanont, N., Protas, H., Jagust, W., Fleisher, A., Reiman, E., Yao, L., Chen, K.: The receiver operational characteristic for binary classification with multiple indices and its application to the neuroimaging study of Alzheimer's disease. IEEE/ACM Trans. Comput. Biol. Bioinf. **10**, 173–180 (2013)
21. Xie, S.X., Baek, Y., Grossman, M., Arnold, M.S., Weiner, M.W., Thal, L.J., Peterson, R.C., Jack, C., Jagust, W., Trojanowski, J.Q., Toga, A.W., Beckett, L.: Alzheimer's disease neuroimaging initiative. Neuroimaging Clin. N. Am. **15**(4), 869–877 (2008)