Addressing a Bottleneck in Data Integration using Automated Learning Techniques

Eduard Hovy, José Luis Ambite, and Andrew Philpot

Information Sciences Institute of the University of Southern California
4676 Admiralty Way, Marina del Rey, CA 90292-6695
{hovy,ambite,philpot}@isi.edu

ABSTRACT

In this paper we describe a new research project that addresses the problem of mapping data across institutions. Given the wide range of geographic scales and complex tasks the Government must administer, its data is stored in numerous formats, which makes it very difficult to integrate, share, or compare. Unfortunately, all approaches to create mappings across comparable datasets require manual effort, which is expensive and time-consuming. Despite some promising work, automatic procedures to create such mappings are still in their infancy, since equivalences and differences manifest themselves at all levels, from individual data values through metadata to the explanatory text surrounding the data collection as a whole. More general methods are required to effectively address this problem.

Viewing the data mapping problem as a variant of the cross-language mapping problem of Machine Translation (MT), we plan to employ the new statistical algorithms developed since 1990 in the MT community to discover correspondences across comparable datasets at all levels. If our automatically learned mappings are effective, we should be able to reduce the amount of manual labor required in database wrapping. We are collaborating with EPA staff at the California Air Resources Board in Sacramento, who periodically integrate data from some 35 regional Air Quality Management Districts throughout California into a single California-wide database, and pass this along to the Federal EPA in North Carolina.

1. INTRODUCTION

When we turn to the Government for information, we expect it to be timely, thorough, and above all accurate. However, due to the wide range of geographic scales and complex tasks the Government must administer, its data is split in many different ways—federal, state, and local; executive, judicial, and legislative; tax management separated from pensions and health, etc.—and it must be collected at different times by different agencies. The result is massive data heterogeneity, expressed especially in incompatible data resources. To deal with this, agencies require appropriate technology. For example, in working with Government agencies, we have found that although relevant information is frequently present and available in sister agencies, no effective technology exists for finding it and converting it into a form usable in-house. The result is that data sharing and integration simply doesn't occur, leading to duplicated data generation efforts in some cases, diminished policy planning effectiveness in others, and wasted public resources in general.

Technology is not the only complicating factor. In many instances, data collected by one agency is regularly used by another, but the consumer has no regulatory or financial means to foster

collaboration. Goodwill, and the amount of time available at the producer and consumer agencies, is often the only way that data transfer can take place.

A third factor is litigation. In our experience, Government agencies usually perform rigorous scrutiny before they release any data, mainly because of fear of litigation by social activists or commercially powerful antagonists. Lacking the tools to perform data filtering and/or transformations easily and effectively, many Government agencies simply make available only a small fraction of the large amounts of data they have laboriously collected. The rest is buried or lost.

The net effect is one of frustration all round. Government officials know they could produce better balanced, more complete reports, but they don't have the technical or other means to. The public, in the form of journalists and scholars, cannot find information that by statute should be available, and when they do, the data are often (and sometimes significantly) inconsistent across multiple agencies. The Legislature likewise cannot get a single and coherent composite picture from the agencies, which leads to misleading information and hence ineffective governance.

This paper describes a new project, trying to address this problem with automated learning techniques used with considerable success in Computational Linguistics. The project is scheduled to start in the summer of 2003, and involves EPA partners working in Air Quality and Fire Emissions Control. Sections 2.1 to 2.4 outline previous experience with the problem and the general approach. Section 2.5 provides some technical details of the planned work.

2. BODY

2.1 Our Previous Work on this Problem

A good example of the data problem appears in the more than 70 US Federal Statistics (FedStats) agencies that collect information about all aspects of life in the USA. Collectively, these agencies have tens of thousands of databases, stored in numerous formats (database software, web pages, typewritten tables, etc.), with new ones being added every day. Frequently, portions of this data overlap, or are semi-complementary (an individual in one database may be part of a family in another, for example). Often, the classes of data are related, near-identical, or even identical (what is termed *salary* in one database, for example, might be exactly the same as *income* in another and *wages* in a third, but might be quite different from what some other agency means by *salary)*.

Some mechanism is required to standardize data types, enable data sharing, and facilitate single-entry-point perusal of all data at hand, regardless of source or type. These desiderata make up a very tall order, for which no simple and wholly complete solution might ever be found. But in previous work we and others have made some progress. Working with representatives from several FedStats agencies, we (at USC/ISI jointly with researchers from Columbia University) built the Energy Data Collection (EDC) system (Ambite et al. 02; Ambite et al. 01). This system, a prototype, enables more dynamic yet homogeneous access to over 50,000 tables of energy data stored in various locations in various formats. The work was funded under the National Science Foundation's Digital Government program in 1999.

2.2 A Core Problem

The EDC system used as data standardization mechanism a large-scale ontology (Hovy 03) and as access tool an AI-based query decomposition planner called SIMS (Arens et al. 96; Ambite and Knoblock 00). Various other approaches have been researched. In all the approaches, a most pressing and fundamental problem is what can be called the *Data Mapping Problem*: identifying equivalent or near-equivalent concepts in different databases, and understanding what the remaining differences are. Without such mappings, one cannot effectively locate, share, or compare data across sources, let alone achieve computational data interoperability.

To date, all approaches involve manual effort when creating these mappings. There is no automated solution to the data mapping problem. But it is endemic to the establishment of data standards and the creation of effective cross-database access mechanisms. Despite some promising work (Doan et al. 00; Muslea et al. 01), the automated creation of such mappings is still in its infancy. Our own work in this regard (Hovy 98; Hovy et al. 01), focusing on cross-ontology alignment in EDC and earlier projects, has been promising but requires larger examples and more powerful training methods. Only with more general methods can one effectively address this problem, since equivalences and differences manifest themselves at all levels, from individual data values through metadata to the explanatory text surrounding the data collection as a whole.

2.3 Our Technical Approach—Overview

We believe the most promising approach available today is to use the new statistical algorithms developed since 1990 in the Machine Translation (MT) community. Following some 35 years of research on MT in Computational Linguistics, a group at IBM Yorktown Heights pioneered a new method in which learning procedures automatically induced cross-language correspondences from 3 million sentences of parallel French and English from the Canadian Parliamentary Hansard records. Various groups have enthusiastically extended these techniques further. One of the most prominent is USC/ISI's MT project, led by Dr. Kevin Knight. Section 2.5.3 discusses new techniques and software packages, including the EGYPT package developed by a team led by Dr. Knight, that have revolutionized MT.

We view the data mapping problem as a variant of the cross-language mapping problem of MT. Instead of mapping French or Arabic words, phrases, and grammatical structures into English ones, we propose to learn the mappings of data from one database to another: using as learning features data values, formats, data subsets, metadata nomenclature and information, additional information such as footnotes and metadata description text, etc. The details are described in Section 2.5.4.

For this work we combine over 10 years' experience with database integration and access, starting with the SIMS project (Arens et al. 96; Ambite and Knoblock 00) and continuing through to the EDC project described above, with access to our MT project, which is a sister project in the same research group at USC/ISI.

2.4 Our Government Partners

In our previous Digital Government project EDC, we collaborated primarily with the Energy Information Administration (EIA), secondarily with the Bureau of the Census and the Bureau of

Labor Statistics (BLS), and peripherally with the National Center for Health Statistics (NCHS). From this experience we have learned the value of working with multiple government partners.

We plan to collaborate closely with several core partners right from the start. In order to help ensure generality, we will develop and test techniques in two different domains. In Phase I we plan to focus on Air Quality Management and in Phase II on Fire Emissions Control. In both phases we will work with appropriate partners in government (who provide the data) and in research (who provide the computational environments and help evaluate the effectiveness of our techniques). In collaboration with our research partners, we propose to adopt a program of graduated expansion, starting with a relatively small core group of government partners and gradually widening the network, as shown in Figure 1.

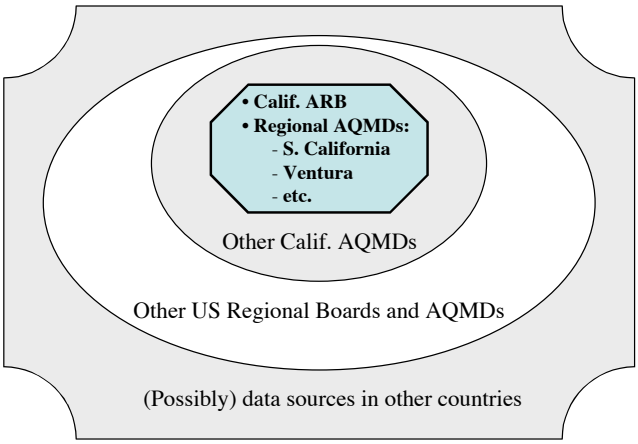

Figure 1. Widening scope of government partnerships, showing only Air Quality partners.

Fortunately, in Phase I, we have an ideal setup, since both the data providers, the California Air Quality Management Districts (AQMDs), *and* the primary data consumer, the California Air Resources Board (CARB), are willing to collaborate. (This is a crucial distinction with our prior EIA project, where we had access only to the data provider, and no method of requesting additional data for broadening the data access and provision channel.)

We have been designing this project with Dr. Michael Benjamin, Manager of the Emission Inventory Systems Section at CARB in Sacramento, CA, who has introduced us to personnel from various of the 35 regional AQMDs in California. Each AQMD collects information on air quality, representing it in a wide variety of formats, from databases to excel spreadsheets. Periodically, personnel from CARB integrate the data into a single California-wide database, and pass this along to the Federal EPA (USEPA) in North Carolina. Even within California, the problems of data integration mentioned above (various formats, autonomous collecting agencies, fear or litigation) exist.

We will initially work with a selected subset of the AQMDs and develop, by hand, data transformation mappings. We do not expect this to be problematic. We will then start developing algorithms to learn these mappings automatically, and apply them to new AQMD data. Subsequently, guided by our Government partners at USEPA, we will search nationwide for comparable data represented in collections and formats that are interestingly different, to exercise and extend our mapping learning algorithms. In Phase II we will work on a different data, namely Fire Emissions, which is produced by a different set of EPA offices, the

USDA/Forest Service, and the Department of Interior. Eventually, and again guided by USEPA, we would like also to consider databases from other countries (an early candidate is Mexico, given its proximity to California and certain factories, transportation routes, etc., near the border).

We will not only develop the learning algorithms but will also package the data mappings for subsequent use by our Government partners. In this we are led by prior experience with the EIA: during the EDC project we collaborated with the startup company Fetch.com (a spinoff from USC/ISI) to build, for the EIA, a software data conversion package that they use in-house in Washington, DC.

2.5 Technical Approach—Details

2.5.1 The Core Problem: Alignment and Transformation

The data mapping problem is the challenge of specifying how the contents of one database map into another. More exactly, the problem is: for a given column (i.e., metadata concept or specification) in the target database, determine the transformation function that applied to one or more column(s) in the source database(s) will produce the target. Since this challenge requires understanding the nature of the denotation (Frege's 'extension') of each column (i.e., column's concept), it is a job that requires humans—usually domain experts.

Computational attempts typically try merely to suggest transformations, or more commonly, a weaker form—alignments—which humans have to then bless or reject. One can approach this problem at various levels. In previous work (Hovy et al. 01; Hovy 98; Knight and Luk 94; Ageno et al. 94; Agirre et al. 94), we and others have focused on simply mapping concepts from one ontology to another, using a function to combine various alignment suggestion heuristics (name match, definition match, etc.) in different weightings. As mentioned above, we have also experimented with applying these heuristics to aligning individual domain terms extracted from domain texts (webpages, manuals, etc.) into a domain ontology (Hovy et al. 01). All these techniques use the fact that each item to be aligned is associated with some text (a name, a definition, etc.).

Others have tried to decompose (in essence, align to a metadata model) semi-structured webpage information such as addresses, catalogue information, etc. (Tejada et al. 01; Muslea et al. 01; Doan et al. 00; Levy and Rousset 96; Knoblock et al. 00; Knoblock et al. 01). Typically these approaches leverage the orthographic and rhetorical structure of this material (telephone numbers have characteristic digit patterns; addresses have typical layout).

To our knowledge, so far, no-one has attempted for data mapping to automatically induce transformation rules for numerical information (functions that for example convert degrees Celcius to Fahrenheit, dollar amounts into yen, etc.). The specialized mathematical packages that learn such transformation are beyond the capability of today's general-purpose alignment tools. We will not focus on this in our work.

Leaving aside purely numerical transformations, we will decompose the data mapping problem into classes as follows.

- Class 1: item-to-item (cell-to-cell) mappings
- Class 2: set-to-set (column-to-column) mappings
- Class 3: multiset-to-multiset (column combination) mappings

At its simplest, class 1 mappings simply seek identity: they require the learning system to recognize that a value in a cell in the target database is identical to the value of a cell in the source database. Typically, this will occur with database key columns, where the other data hinges on items such as employee name, subdistrict/region, month, etc. Slightly more complex mappings occur when the cell value has been trivially transformed, such as area codes in phone numbers being parenthesized in one database but not in the other, or "street" and "avenue" being abbreviated in one database and spelled out in the other, etc. Recognizing that these mappings are appropriate can be achieved by having the learning algorithm consider not only the actual cell values but additional features about the cell values, including orthographic ones (spelling and abbreviation patterns, numerical/currency/etc. formats, parenthesis, hyphenation, and other punctuation patterns, etc.).

By itself, a class 1 mapping means very little. Only when (enough of) a whole column in one database is mapped to (enough of) a whole column in another can one postulate that a regular transformation has been applied. Class 2 mappings occur when enough class 1 mappings have been found within a single structural unit (set) such as a column. The operational parameters here are: what is "enough"? What about source cells that do not transform correctly, or target cells for which there is no source? (In some cases, the class 1 transformation applied may be too weak to learn the full mapping; in others, it may have overgeneralized and hence be wrong.) These and similar parameters will be empirically determined.

Class 2 mappings are also often not sufficient. A target column may appear to be produced by two source columns, for example. (This will certainly be the case if the set of class 1 transformations are very weak, for example simply recognize *number*□ *number*.) Class 3 algorithms are required to compare the putative mappings at the level of class 2, either to resolve competing alternatives or to find joint input (composed mappings). The latter case is difficult, and we will refer to database theory on the topic of Joins to define a set of simple composition operators that appear in our domains.

2.5.2 Some Example Transformation Mappings

Consider two information sources I1 and I2, with tables T1 and T2, which have columns C1 and C2, respectively. Creation of a mapping between T1 and T2 means generating and testing hypotheses about elements of the two sources. Some of the particular mappings that we may encounter include:

- T1 _ T2 (identity). T1 and T2 provide precisely the same information; each column has an identical analogue in the other table. T1 is essentially a replicated T2.
- T1 = T2 (class identity). This is a weaker relationship. For example, T1 and T2 might both represent each of the 50 US states, and yet T1 might list state capitals and T2 populations.
- T1 <_ T2 and T1 <= T2 (subset, subclass identity). T1 could contain the western states while T2 contains all states. The relation <_ indicates proper subset of the rows of the larger relation; <= indicates instead a subset of the extended objects with possibly different attributes.
- T1.C1 = T2.C2 (attribute similarity; object identity when C1, C2 are keys). All values in T1.C1 are found in T2.C2 and vice versa. For example, T1 contains all US cities by state; T2 contains US counties by state.

- T1.C1 <= T2.C2 (attribute subset; foreign key when C2 is a key of T2). A subset of values of T1.C1 is in T2.C2. For example, T1 contains cities in the US, by state; T2 contains all zip codes of all states.

We may also generate and test weaker hypotheses about single or multiple tables and columns, such as:

- T1.C1 <= known syntactic type (type membership). For example, each T1.C1 is a three-digit integer greater than 200 and less than 1000 with middle digit never '9'; hence, a possible US area code.
- T1.C1 <= standard range (range membership) For example, each T1.C1 is between 0.0 and 100.0, possibly a nominal percentage.
- T1.C1 although declared nullable is never NULL.
- type(T1.C1) _ type(T2.C2) (type equivalence). By this we mean they have same declared database column type e.g., both are VARCHAR(31). A weaker version where both are of compatible types (e.g., both are VARCHAR) may be useful as well (data type compatibility).
- name(C1) _ name(C2) (name match). Substring name match and name equivalence modulo terminological reference (*salary* vs. *wages*) or NL translation (*money* vs. *argent*) are also useful when feasible.

In each case, we may use probabilistic measures based on partial sampling when data noise or size preclude full analysis.

Several of these mapping hypotheses refer to metadata level attributes of tables and/or columns. At least three kinds of metadata will be useful in deriving mapping relationships between source descriptions:

- Definitional metadata. Accompanying formal metadata, associated documentation, and column name can all be useful in discovering column semantics and linking columns (e.g., C1 is "PhoneNum").
- Implementation metadata (declared type, declared key, nullable etc.) (For example, T1.C1 is declared an INTEGER(7) and might contain a US local phone number.
- Derived meta-level attributes. We can perform selected analyses (complete or sampled) on certain columns to determine likely patters. For example, if T1.C1 always contains a 7-digit number greater than 2000000, it might represent a US local phone number.

1.1.3 Recent Machine Translation Research on Alignment and Mapping

Which algorithms will we employ to actually perform the learning? Setting source and target databases side by side, the problem resembles that of alignment in machine translation research. In the early 1990s, a group at IBM (Brown et al. 93) cast translation in the theoretical framework of the Noisy Channel Model (used in speech recognition and telephone communication engineering), using Bayes' Rule:

$$P(E|F) = \text{argmax } P(F|E) \cdot P(E)$$

(the probability of producing an English word/phrase/etc. from French input is the overall maximum of the multiplied probabilities of producing the French out of the English and the probability of the English by itself in the first place). To learn the former probabilities they used 3 million sentences of French and English, parallel, from the Canadian Parliamentary records. To train the latter probabilities, approximated by bigrams (and later trigrams), they used several years of the *Wall Street Journal*. The former term, $P(F|E)$, they called the translation model, and

it recorded the likelihood that any word in English would be translated into a word in French. In later models, they added refinements: fertility (the likelihoods that a word would produce from 0 to 5 translations), distortion (where in the sentence translation words would appear), etc. This work pioneered the use of Expectation Maximization in Computational Linguistics.

Since then, several other approaches to character, word, and sentence alignment have been developed, including Charalign (Gale and Church 91) and the algorithms of Melamed (00). The former uses constant anchor points such as paragraph breaks and punctuation; the latter's use of a binomial distribution parameter makes computation rather complex and slow. Space limitations prevent further discussion.

In 1999, our colleague at USC/ISI, Kevin Knight, led a summer team at Johns Hopkins in building a refined and very fast set of algorithms to learn the correspondences and perform statistics-based machine translation (Knight et al. 99). Their toolkit EGYPT is freely available from the web at http://www.clsp.jhu.edu/ws99/projects/mt/toolkit/. In the kit, Whittle is a software tool for preparing and splitting bilingual corpora into training and testing sets, GIZA is a training program that learns statistical translation models from bilingual corpora, cairo is a word alignment visualization tool, and cairoize is a tool for generating alignments files in the format required by cairo.

Though as mentioned earlier the data mapping problem is not identical to the bilingual alignment problem, it can be formulated in a way that some of the MT tools can be employed. In particular, we will use GIZA, cairoize, and cairo, suitably adapted to the EPA data.

1.1.4 Proposed Learning Algorithms and Procedure

How will we learn the mapping transformations? Treating each database cell as the analogue of a word in EGYPT, we will begin with a set of very simple class 1 transformation functions, including *number* *number*, *word* *word*, etc. Led by analysis of the data, we will manually add new transformation functions (such as *7digits* *add-parens-at1and3* and other variations of the types listed in Section 2.5.2 above) to be able to recognize increasingly complex transformations, recording with each its typicality in the domain, its expected utility (a function of its accuracy as determined by the human judge), etc. Each such transformation function will license certain potential mappings, which will be made available to GIZA using a modified form of Whittle. We will visualize GIZA's results using cairo and ask the human expert to reject or to bless the mappings found by GIZA to be statistically most likely. This feedback will be incorporated into the rating of each transformation function. Ultimately, we will produce a collection of transformation functions, some operating on cell values, some on cell value orthography, some on cell value formatting, etc., and each with its rating/utility weight. This collection will constitute the bulk of the learning algorithms of class 1.

For class 2, we will treat each column as a unit (equivalent now to a word in EGYPT). The set of possible alignments for all the cells in the column will constitute the alignment ambiguity set of the column as a whole. As before we will define a collection of class 2 transformation functions, beginning with the simplest (identity, if a threshold number of cell alignments agree), and eventually considering metadata specifications such as column heading and definitional agreement, using the name and definition match heuristics (Hovy 98). As before we will use GIZA to learn the best alignments under various sets of transformation functions, and rate the functions.

While we will apply the same procedure for class 3 mappings, we will also be alert to the possibility of using simpler procedures. Given the much smaller amount of training data at this level and the increased complexity of transformation functions, it may be most effective to apply a mathematical function induction package, if simple analysis decides that a composition of transformation functions is likely (if, for example, multiple source columns seem to map to a target column but no obvious redundancy is found, or if a target column is not apparently generated by any source columns at all, in other words comes out of thin air, which is of course impossible). With sufficient time we may investigate this.

The sets of transformation functions at each level, and the scripts to connect them, constitute the final mapping package for a given source database and a given target database.

1.6 Evaluation

We will perform both extrinsic and intrinsic (Spärck Jones and Galliers 96) evaluation during the project.

Intrinsic evaluations (glass-box) measure the performance of components of a system compared to previous versions of it, and are primarily useful to the developer. We will periodically measure the accuracy (Precision, Recall, sensitivity to training set size, and other standard learning measures) of each of the transformation functions we employ. We will compare these scores to the typical and best performances of the techniques reported for other applications, in such forums as the journal *Machine Learning* and the conferences EMNLP and SIGDAT.

Extrinsic evaluations (black-box) measure the functionality of the system when employed for some task, and are mostly informative to the user. We plan to collaborate with both our research partners in testing the effectiveness of the results of our learning algorithms. Under controlled conditions, and taking into account human variation such as expertise with the process and domain knowledge, this involves comparing the speed with which humans who perform database wrapping do their job: either in their traditional (fully manual) mode, or using the mappings learned by our system. One has to conduct enough trials to account for the variability introduced by humans, since we cannot use the same person to do the mapping twice, of course.

The results of extrinsic evaluations are not only gross time difference numbers. We will also interview the human to elicit their experiences and suggestions, and use these in subsequent stages of our work, in particular in the formatting and presentation of the learned mappings.

The ultimate extrinsic evaluation, of course, is the end user. Our dream is to build a toolkit that allows them simply to pour in enough source and target data and to get an automated mapping system out.

3. CONCLUSION

This work will apply emerging statistical techniques for machine translation (MT) to the problem of automating database schema integration. In MT, the techniques align words and word sequences across languages. This research will adapt and extend the techniques to consider not only data values (the analogue of words) but also data format/orthography, metadata information, and associated textual information (metadata descriptions, footnotes, etc.) in the alignment process, and to perform alignment learning at three levels: individual data cell level, set of cells (column) level, and multi-column level. Multi-level alignment has not been attempted in MT

before. These powerful learning techniques have never been applied to metadata schema integration and/or database alignment or wrapping.

To the extent this work succeeds, it has the potential to significantly reduce the amount of human work involved in creating single-point access to multiple heterogeneous databases. This problem is faced by thousands of large enterprises with numerous data collections, from Government agencies at all levels to the chemical and automotive industries to startup companies that link together and integrate websites. By automatically postulating mappings across databases/metadata, the proposed algorithms will enable the database wrapper builder (whether fully manual or semi-automated) to work more quickly and effectively. It will also help with the creation of metadata standards.

In particular, we will provide our results to our partner agencies in the EPA so that they can transform their data at will. Working with our partners at the Federal EPA, we also plan after the first year to work on mapping appropriate data collections of other US states and eventually, possibly, other countries (such as Mexico).

4. REFERENCES

Ageno, A., I. Castellon, F. Ribas, G. Rigau, H. Rodriguez, and A. Samiotou. 1994. TGE: Tlink Generation Environment. *Proceedings of the 15th COLING Conference*. Kyoto, Japan.

Agirre, E., X. Arregi, X. Artola, A. Diaz de Ilarazza, and K. Sarasola. 1994. Conceptual Distance and Automatic Spelling Correction. *Proceedings of the Workshop on Computational Linguistics for Speech and Handwriting Recognition*. Leeds, England.

Ambite, J.L. and C.A. Knoblock. 2000. Flexible and Scalable Cost-Based Query Planning in Mediators: A Transformational Approach. *Artificial Intelligence*, 118(1-2):115–161.

Ambite, J.L., Y. Arens, E.H. Hovy, A. Philpot, L. Gravano, V. Hatzivassiloglou, J.L. Klavans. 2001. Simplifying Data Access: The Energy Data Collection Project. *IEEE Computer* 34(2).

Ambite, J.L., Y. Arens, W. Bourne, P.T. Davis, E.H. Hovy, J.L. Klavans, A. Philpot, S. Popper, K. Ross, J.-L. Shih, P. Sommer, S. Temiyabutr, L. Zadoff. 2002. A Portal for Access to Complex Heterogeneous Information about energy. *Proceedings of the dg.o 2002 Conference.* Los Angeles, CA. May 2002.

Arens, Y., C.A. Knoblock and C.-N. Hsu. 1996. Query Processing in the SIMS Information Mediator. In A. Tate (ed), *Advanced Planning Technology*. Menlo Park: AAAI Press.

Brown, P.F., V.J. Della Pietra, S.A. Della Pietra, and R.L. Mercer. 1993. The Mathematics of Statistical Machine Translation: Parameter Estimation. *Computational Linguistics* 19(2).

Doan, A., P. Domingos, and A. Levy. 2000. Learning Source Descriptions for Data Integration. *Proceedings of a Workshop at the Conference of the American Association for Artificial Intelligence*.

Gale, W.A. and K.W. Church. 1991. A Program for Aligning Sentences in Bilingual Corpora. *Proceedings of the ACL Conference*.

Hovy, E.H. 1998. Combining and Standardizing Large-Scale, Practical Ontologies for Machine Translation and Other Uses. *Proceedings of the 1st International Conference on Language Resources and Evaluation* (LREC). Granada, Spain.

Hovy, E.H., A. Philpot, J.L. Ambite, Y. Arens, J. Klavans, W. Bourne, and D. Saroz. 2001. Data Acquisition and Integration in the DGRC's Energy Data Collection Project. *Proceedings of the NSF's National Conference on Digital Government dg.o 2001.* Los Angeles.

Hovy, E.H. 2003. Using an Ontology to Simplify Data Access. *Communications of the ACM* Special Issue on Digital Government. January.

Knight, K. and S.K. Luk. 1994. Building a Large-Scale Knowledge Base for Machine Translation. *Proceedings of the AAAI Conference*.

Knight, K. et al. 1999. EGYPT MT Toolkit. http://www.clsp.jhu.edu/ws99/projects/mt/toolkit/.

Knoblock, C.A., K. Lerman, S. Minton, and I. Muslea. 2000. Accurately and Reliably Extracting Data from the Web: A Machine Learning Approach. *Data Engineering Bulletin* 23(4).

Knoblock, C.A., S. Minton, J.L. Ambite, N. Ashish. I. Muslea, A.G. Philpot, and S. Tejada. 2001. The Ariadne Approach to Web-based Information Integration. *International Journal on Cooperative Information Systems* (IJCIS) 10(1–2) Special Issue on Intelligent Information Agents: Theory and Applications (145–169).

Levy A.Y. and M.-C. Rousset. 1996. Carin: A representation language integrating rules and description logics. *Proceedings of the European Conference on Artificial Intelligence*, 323–327. Budapest, Hungary.

Melamed, I.D. 2000. Models of Translational Equivalence among Words. *Computational Linguistics* 26(2): 221–249.

Muslea, I., S. Minton, and C.A. Knoblock. 2001. Hierarchical Wrapper Induction for Semistructured Information Sources. *Journal of Autonomous Agents and Multi-Agent Systems* 4:93–114.

Spärck Jones, K. and J.R. Galliers. 1996. Evaluating Natural Language Processing Systems: An Analysis and Review. New York: Springer.

Tejada, S., C.A. Knoblock, and S. Minton. 2001. Learning Object Identification Rules for Information Integration. *Information Systems* 26(8).

5. KEY WORDS

Data conversion

Data mapping

Automated learning of data mapping

Machine learning techniques for data mapping