Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence

Entity Resolution in a Big Data Framework

Mayank Kejriwal
University of Texas at Austin
5.424C, Stop D9500 2317 Speedway, Austin, TX 78712
mayankkejriwal.azurewebsites.net
kejriwal@cs.utexas.edu, 1-217-819-6696

Resource Description Framework (RDF)[1] is a data model that can be used to publish semistructured data visualized as *directed graphs*. An example is Dataset 1 in Fig. 1. Nodes in the graph represent *entities* and edges represent *properties* connecting these entities. Two nodes may refer to the same *logical* entity, despite being syntactically disparate. For example, the entity *Mickey Beats* in Dataset 1 is represented by two syntactically different nodes.

Entity Resolution (ER) is the problem of resolving such semantically equivalent entities by linking them using a special *sameAs* property edge (Ferraram, Nikolov, and Scharffe 2013). The ER problem is not restricted to the RDF data model but can be stated *abstractly* as identifying and resolving semantically equivalent entities in one or more datasets (Elmagarmid, Ipeirotis, and Verykios 2007). As an example of applying ER on *relational* datasets, consider Datasets 2 and 3 in Fig. 1. In these datasets, an entity is represented as a *tuple*. The goal is to identify *duplicate tuples*, that is, tuples referring to the same logical entity.

ER is an important AI problem that has been acknowledged as occurring in structured, semistructured and even unstructured data models. A survey on the subject cites at least eight different names for the problem, including record linkage, instance matching, link discovery and co-reference resolution (Elmagarmid, Ipeirotis, and Verykios 2007).

The problem has grown in concert with the Semantic Web and with the publishing of new data on the Web. Given the prevalence of large datasets in data integration applications (Goodhue, Wybo, and Kirsch 1992), an ER solution must be *scalable*. For example, consider *Linked Open Data* (LOD[2]), which is the collection of RDF datasets published under an open license (Bizer, Heath, and Berners-Lee 2009). LOD currently contains over 30 billion triples and over 500 million property edges, published in over 300 datasets. Studies have suggested that LOD contains many syntactically disparate but semantically equivalent entities that have not yet been discovered and linked (Papadakis et al. 2010). In the relational domain, the *Deep Web*, which is the collection of back-end relational databases powering Web queries and faceted search, has also shown super-linear growth and is at least 500 times greater than the Surface Web, according to a study (He et al. 2007).

ER in support of co-referencing data *across* LOD and Deep Web mandates meeting a *heterogeneity* requirement. Consider Fig. 1 again. *Mickey Beats* is present as a logical entity in all three datasets. An ER system meeting the heterogeneity requirement would be flexible enough to link entities across both data models, and would not have to be configured for each model individually.

Given the expense of domain expertise, an ER system should minimize supervision and be *automated*. It would be an added boon for the system to be *cloud*-deployable, since it could then be accessed as a service over the Internet. ER applications have been widely documented, with examples including populating an Entity Name System (ENS) for enabling semantic search (Bouquet and Molinari 2013) and improving accuracy in identifying knowledge graphs (Pujara et al. 2013). An ER cloud service would benefit such efforts, and potentially allow others to expand to Web-scale.

Current state-of-the-art is unable to meet the motivated requirements of heterogeneity, scalability and automation *simultaneously*, as exhaustive surveys on the subject show (Christen 2012). The thesis is that building such a system requires identifying and resolving a novel set of challenges that have been heretofore unacknowledged.

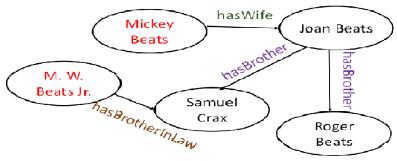
This dissertation presents a fully unsupervised ER prototype, the key components of which are implemented in the *MapReduce* framework (Dean and Ghemawat 2008). In the general case, the prototype accepts $N \geq 1$ heterogeneous databases as input, and outputs a set of matched entities, which may be used by subsequent applications. A high-level schematic is shown in Fig. 2. Current ER systems are designed only for $N = 1$ or $N = 2$ (Elmagarmid, Ipeirotis, and Verykios 2007). Enabling a workflow for arbitrary N, which we designate the *N-Way* problem, is currently an open area of research and involves novel challenges.

We evaluate all proposed algorithms both on established benchmarks, as well as *new* datasets procured in the hope of aiding future research efforts. We implement the prototype on 32 HDInsight[3] nodes in the Microsoft Azure cloud infrastructure and evaluate it on real-world Big Data.

[1] http://www.w3.org/RDF/

[2] linkeddata.org

[3] http://azure.microsoft.com/en-us/services/hdinsight/. These nodes are designed to facilitate Apache *Hadoop* as a service



Dataset 1

Emergency Contact

Name	Contact	Relation
Mickey Beats	Joan Beats	Spouse
Susan Palermo	None	*null*
Samuel Crax	Joan Beats	Sister

Dataset 2

Personal Details

First Name	Last Name	Phone	Zip
Mickey	Beats	333-310-4400	77440
Susan	Palermo	*null*	65764
Samuel	Crax	217-413-4990	69874
Mickey	W. Beats Jr.	310-4400	77440

Dataset 3

Figure 1: An instance of heterogeneous *Entity Resolution*, with *Mickey Beats* needing to be *resolved* across 3 databases

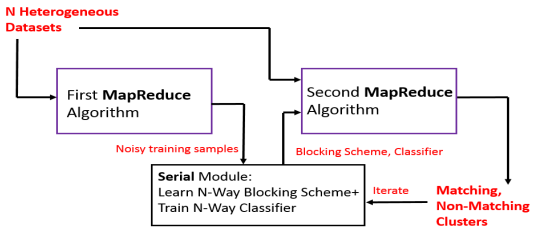


Figure 2: The N-Way ER prototype developed in the thesis

Table 1: An overview of the criteria that contributions and proposed work fulfill. A, H, G and S stand for *Automation*, *Heterogeneity*, *Scalability* and the *N-Way* problem respectively

Work	A	H	S	N	Status
Contribution 1	Yes	No	No	No	Done
Contribution 2	Yes	Yes	No	No	Done
Contribution 3	Yes	No	Yes	No	Done
Proposed Work 1	Yes	Yes	Yes	No	4 months
Proposed Work 2	Yes	Yes	Yes	Yes	6 months

A summary of completed and proposed contributions (as of September, 2014) is presented as a matrix in Table 1. For the proposed work, an approximate time frame for completion is indicated. Note that some of the contributions span multiple publishable papers. An example is Contribution 2, which adds *heterogeneity* to Contribution 1. In two separate works, we apply our proposed techniques to different parts of the ER pipeline for RDF inputs. One of these has been accepted for a workshop publication at ISWC[4], while the other is currently pending as a regular paper at ACM SIGMOD. For full details on all these publications, please refer to the attached Curriculum Vitae. We also plan to integrate these heterogeneity efforts into a December journal submission.

By the time of the AAAI conference, I expect to have made refinements to Contribution 2 efforts, by way of multiple channels of feedback from reviewers and conference attendees, and to have submitted a Contribution 3 paper to a data mining conference. Considerable progress on Proposed Contribution 1 is anticipated, but the full effort is expected to reach fruition only after the conference ends, after which Proposed Contribution 2, a relatively open research area, will become a full-time priority. I also expect to spend 3 months writing the dissertation and repeating experiments.

The principal project investigators on all described efforts are myself and my advisor, Daniel P. Miranker. In papers describing completed work, we are currently the only authors. I would also like to acknowledge a graduating Ph.D. student, Juan Sequeda, for his generous expertise on Semantic Web technology and data mapping at key points of the research.

[4]International Semantic Web Conference: iswc2014. semanticweb.org

References

Bizer, C.; Heath, T.; and Berners-Lee, T. 2009. Linked data-the story so far. *International journal on semantic web and information systems* 5(3):1–22.

Bouquet, P., and Molinari, A. 2013. A global entity name system (ens) for data ecosystems. *Proceedings of the VLDB Endowment* 6(11):1182–1183.

Christen, P. 2012. Further topics and research directions. In *Data Matching*. Springer. 209–228.

Dean, J., and Ghemawat, S. 2008. Mapreduce: simplified data processing on large clusters. *Communications of the ACM* 51(1):107–113.

Elmagarmid, A. K.; Ipeirotis, P. G.; and Verykios, V. S. 2007. Duplicate record detection: A survey. *Knowledge and Data Engineering, IEEE Transactions on* 19(1):1–16.

Ferraram, A.; Nikolov, A.; and Scharffe, F. 2013. Data linking for the semantic web. *Semantic Web: Ontology and Knowledge Base Enabled Tools, Services, and Applications* 169.

Goodhue, D. L.; Wybo, M. D.; and Kirsch, L. J. 1992. The impact of data integration on the costs and benefits of information systems. *MiS Quarterly* 293–311.

He, B.; Patel, M.; Zhang, Z.; and Chang, K. C.-C. 2007. Accessing the deep web. *Communications of the ACM* 50(5):94–101.

Papadakis, G.; Demartini, G.; Fankhauser, P.; and Kärger, P. 2010. The missing links: Discovering hidden same-as links among a billion of triples. In *Proceedings of the 12th International Conference on Information Integration and Web-based Applications & Services*, 453–460. ACM.

Pujara, J.; Miao, H.; Getoor, L.; and Cohen, W. 2013. Knowledge graph identification. In *The Semantic Web–ISWC 2013*. Springer. 542–557.