

## Project Proposal: The Disambiguation Problem

Industrial Partner: Lawrence Livermore National Laboratory

Industrial Adviser: Tina Eliassi-Rad ([eliassi@llnl.gov](mailto:eliassi@llnl.gov))

Faculty Mentor: Artur Szlam ([arthur.szlam@yale.edu](mailto:arthur.szlam@yale.edu))

### 1 Project Description

Disambiguation deals with the problem of detecting and consolidating *redundant data*<sup>1</sup> (such as data with misspellings, missing information, or invalid information). Disambiguation is essential for maintaining data quality and has been investigated in a variety of fields such as artificial intelligence and databases. Two major challenges for this problem are *accuracy* (e.g. AUC<sup>2</sup>) and *speed* (e.g. the number of data points consolidated per second). Of course, values for accuracy and speed depend not only on the chosen approach but also on the size of data and complexity of the concept being disambiguated. We frame the disambiguation problem as follows:

Inputs:

- A directed graph  $G = (V, E)$ , where  $G$  has the following properties:
  - *Dynamic*: Data is continuously being added to  $G$ .
  - *Labeled*: Vertices and edges (namely,  $V$  and  $E$ ) have heterogeneous attribute vectors associated with them. A *key* attribute for each vertex and each edge is *type* (such as *person* for a vertex and *authored* for an edge).
  - *Multi-modal and multi-source*:  $G$  contains data with various representations from multiple sources.
- A subgraph  $S = (V_S, E_S)$ , where  $|\{V_S\}| \geq 1$  and  $|\{E_S\}| \geq 0$ .

Output:

- A subgraph  $R = (V_R, E_R) \in G$ , where  $\mathop{\text{arg max}}_R \text{Similarity}(R, S)$ .
  - *Similarity* is a function that measures the “closeness” between subgraphs  $R$  and  $S$ . *Similarity* has to capture both structural (graph-based) similarity and semantic (attribute-based) similarity. The performance of any solution to the disambiguation problem will (obviously) depend heavily on the chosen *Similarity* function.<sup>3</sup>

### 2 Background

The disambiguation problem has been studied in a variety of areas:

- In data mining, it is part of *data cleaning* and is called *record linkage*, *object consolidation*, *deduplication*, *entity identification*, or *entity reconciliation* [1][9].
- In artificial intelligence, it is called *identity uncertainty*, *object identification*, *object correspondence*, *reference matching*, or *alias detection* [2][5][6].

The complex networks literature is another source of background information. Of particular interest are publications on *community structures* [7][8] and *vertex similarity* [4].

---

<sup>1</sup> We define redundant data as multiple instances of data that refer to the same object.

<sup>2</sup> AUC stands for area under the curve, where the curve refers to ROC (receiver operating characteristics).

<sup>3</sup> Deciding when and how to consolidate two subgraphs are beyond the scope of this project.

A common approach to solving disambiguation includes the following two steps:

- I. *Block*, where vertices and edges in  $G$  that are obviously not similar to  $S$  are pruned. This step usually includes building an index on  $G$ , which quickly allows detection of potential redundancies. An effective *Similarity* function can be used to generate a promising index. For example, a clustering algorithm [3] can utilize such a function to group/cluster similar subgraphs in  $G$  and generate an index on it.
- II. *Match*, where a *Similarity* function is used to evaluate a potential match for  $S$  in  $G$ . *Similarity* functions can be learned from the data. A popular approach is to construct probabilistic models that assign *confidence rankings* or *probability of a match* to the potential redundancies. It is important to note that we cannot assume to have annotated/training data for learning the *Similarity* function. So, an unsupervised learning algorithm is preferred to a supervised one.

### 3 Research Tasks and Deliverables

The research tasks focus on the similarity function. In particular, the three main tasks include:

- Task 1: Evaluate a series of *Similarity* functions.
- Task 2: Prototype the most promising approach.
- Task 3: Test its performance on disambiguating people in the *DBLP Computer Science Bibliography* data set [10] using the following performance metrics: AUC and the number of data points consolidated per second.

The deliverables include (1) a technical report detailing approaches taken and lessons learned and (2) the prototype software used to conduct performance tests.

### 4 References

- [1] P. Christen and K. Goiser, "Assessing Deduplication and Data Linkage Quality: What to Measure?" In *Proc. of the 2005 Australian Conf. on Data Mining*, Sydney, Australia, 2005.
- [2] A. Culotta and A. McCallum, "Practical Markov Logic Containing First-order Quantifiers with Application to Identity Uncertainty," Technical Report IR-430, University of Massachusetts, September 2005.
- [3] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data Clustering: A Review," *ACM Computing Surveys*, 31(3):264-323, 1999.
- [4] E. A. Leicht, P. Holme, and M. E. J. Newman, "Vertex Similarity in Networks," *Phys. Rev. E*, 73:026120, 2006.
- [5] X. Li, P. Morie, and D. Roth, "Semantic Integration in Text: From Ambiguous Names to Identifiable Entities," *AI Magazine: Special Issue on Semantic Integration*, AAAI Press, 2005.
- [6] B. Milch, B. Marthi, S. Russell, D. Sontag, D. L. Ong, and A. Kolobov, "BLOG: Probabilistic Models with Unknown Objects," In *Proc. of the 2005 Int'l Joint Conf. on Artificial Intelligence*, Edinburgh, Scotland, 2005.
- [7] M. E. J. Newman, "Finding Community Structure in Networks using the Eigenvectors of Matrices," *Phys. Rev. E*, 2006 (submitted).
- [8] M. E. J. Newman, "Modularity and Community Structure in Networks," *PNAS USA*, 2006 (in press).
- [9] Parag and P. Domingos, "Multi-Relational Record Linkage," In *Proc. of the 2004 SIGKDD Workshop on Multi-Relational Data Mining*, Seattle, WA, 2004.
- [10] University of Massachusetts' database of the *DBLP Computer Science Bibliography*, <http://kdl.cs.umass.edu/data/dblp/dblp-info.html>.