Exploiting Secondary Sources for Unsupervised Record Linkage

Martin Michalowski, Snehal Thakkar, and Craig A. Knoblock

University of Southern California
Information Sciences Institute,
4676 Admiralty Way
Marina del Rey, CA 90292 USA
{martinm, thakkar, knoblock}@isi.edu

Abstract

XML, Web services, and the Semantic Web have opened the door for new and exciting information integration applications. Information sources on the web are controlled by different organizations or people, utilize different text formats, and have varying inconsistencies. Therefore, any system that integrates information from different data sources must identify common entities from these sources. Data from many online sources does not contain enough information to accurately link the records using state of the art record linkage systems. There is an inherent need for learning in these systems, most of the time requiring a user in the loop, to accurately link records across datasets. In this paper we describe a novel approach to exploiting additional data sources to design an unsupervised record linkage method. Our evaluation using real world data sets shows that the performance of unsupervised learning in a record linkage system is on par with traditional supervised learning methods.

1 Introduction

In the recent past, researchers have developed various machine learning techniques such as SoftMealy [7] and Stalker [13] to easily extract structured data from various web sources. Using those techniques, users can build wrappers that allow them to easily query web sources much like databases. Web-based information integration systems such as Information Manifold [15], InfoMaster [4], and Ariadne [9] can provide a uniform query interface for users to query information from various web sources as well as databases. While the above-mentioned systems can integrate information from various data sources, none of them completely address the issues relating to textual inconsistencies across several data sources. For example, two restaurant web sites may refer to the same address using different textual information. Therefore, record linkage is essential to accurately integrate data from various data sources.

There has been some work done on linking records from various web-sites using textual similarities and transformations [1, 2, 3, 17]. These approaches provide better consolidation results when compared to exact text matching techniques in different application domains. However, all of these systems rely on a fair amount of user interaction, whether it be in labeling match pairs [1, 17], creating reference and input tables [2], or designing domain specific profilers using a domain expert's knowledge [3]. By incorporating additional data sources into the loop, we can use them to label record pairs. There are many application areas where information from additional data sources can provide important domain knowledge. Examples include utilizing a geocoder to determine if two addresses are the same, utilizing historical area code changes to determine if two phone numbers are the same, and utilizing the location and officers information for different companies to determine if two companies are the same.

The goal of our research is to provide a framework for accurately linking records across data sources in an unsupervised manner. In our previous work [11, 10], we showed how primary sources can be augmented with data obtained from secondary sources to improve the record linkage process. In this paper, we present an extension to Apollo's active learning component to

Proceedings of the 30th VLDB Conference, Toronto, Canada, 2004

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
2004		**00-00-2004 to 00-00-2004**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Exploiting Secondary Sources for Unsupervised Record Linkage	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **University of Southern California,Information Sciences Institute,4676 Admiralty Way,Marina del Rey,CA,90292**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
The original document contains color images.

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**		**6**	

Dinesite	Zagat's	Matched
Broadway Bar and Grill, 1460 3rdstreet Promenade, Santa Monica, CA 90401-2322, 310.393.4211	Broadway Bar&Grill, 1460 Third St. Promenade, Santa Monica, CA, 90401-2322, (310) 393-4211	
C & O Cucina, 3016 Washington Boulevard, Marina del Rey, CA 90292-5549 , 310.301.7278	C & O Tratorria, 31 Washington Boulevard, Marina del Rey, CA 90292, (310)823-9491	

Figure 1: Textual Inconsistencies Across Data Sources

address the issue of user involvement. Using secondary sources, a system can autonomously answer questions posed by its active learning component. By diverting questions to the system itself, the entire record linkage process minimizes the involvement of a user. In the extended Apollo system, user involvement is limited to a preprocessing step used to evaluate secondary sources.

In presenting our approach, we will provide a motivating example, followed by some background work on record linkage. Subsequently, we present an analysis of our methodology for unsupervised learning using secondary sources. We will then present the evaluation of our approach using real world restaurant data sets with both supervised and unsupervised learning techniques. Finally, we will discuss related work and put forward our conclusions and planned future work.

2 Motivating Example

To clarify the concepts presented in this article, we will define the following terms: (1) Record Linkage, (2) Primary data sources, and (3) Secondary data sources. Record linkage is the process of determining if two records refer to the same entity. A primary data source is one of the two initial data sources used for record linkage. A secondary data source is any source, other then a primary data source that can provide additional information about entities in the primary data sources. Consider the following primary data sources:

- Zagat and Dinesite data sources that provide information about various restaurants.
- Travelocity and Orbitz data sources that provide information about various hotels.
- Yahoo and Moviefone data source that provides information about various theaters.

When the user sends a request to obtain information pertaining to restaurants within a given city, the record linkage system needs to link records that refer to the same restaurant from the Zagat and Dinesite data sources. However, due to the textual inconsistencies present in both data sources, determining which records refer to a common entity is a non-trivial task. Figure 1 shows the varying textual inconsistencies found in the restaurant data sources. A similar situation arises when attempting to combine information about hotels from Travelocity and Orbitz, or about movies from Yahoo and Moviefone.

Furthermore, when linking records across data sources, it is desired that a system be able to perform this task as autonomously as possible. It is tedious for a user to label a large number of record pairs as matches or non-matches and desirable for a system to accomplish this task with minimal involvement from the user. However, for such a system to be successful, it must be as good as if a user was in the loop. Therefore, it is important to maintain accuracy while introducing autonomy. This accuracy is maintained by using specialized matching techniques found in secondary sources rather then encoding all possible matching techniques in the system itself.

3 Previous Work

In this section, we present the Active Atlas [16, 17] and Apollo [11, 10] record linkage systems. Active Atlas is used as a foundation upon which the Apollo system is built. Apollo automatically augments primary sources with secondary source information to improve the record linkage process. Its robust extendable framework make it an ideal candidate for a base system upon which to build an unsupervised record linkage system.

3.1 Active Atlas Overiew

Active Atlas' architecture consists of two separate components: a candidate generator and a mapping learner. Its goal is to find common entities amongst two record sets from the same domain. The candidate generator proposes a set of potential matches

based on the transformations available to the system. The transformation may be one of a number of string comparison types such as equality, substring, prefix, suffix, stemming, or others and are weighted equally when computing similarity scores for potential matches. Once the candidate generator has finished proposing potential matches, Active Atlas moves on to the second stage and uses the potential matches as the basis for learning mapping rules and transformation weights.

The mapping learner establishes which of the potential matches are correct by adapting the mapping rules and transformations weights to the specific domain. Due to the fact that the initial similarity scores are very inaccurate, the system uses an active learning approach to refine and improve the transformation weights and mapping rules. This approach uses a decision tree committee model for learning with three members in the committee. The mapping learner selects the most informative potential match and asks the user to label this example as either a match or non-match. The users response is used to refine and recalculate the transformation weights, learn new mapping rules, and reclassify record pairs. This process continues until: (1) the committee learners converge and agree on one decision tree, or (2) the user has been asked a pre-defined number of questions. Once the mapping rules and transformation weights have been learned, Active Atlas uses them to classify all the potential matches in the system as matched or not matched. The results are then made available to the user.

3.2 Automatically Augumenting Primary Data Sources

Current record linkage systems [1, 3, 6, 8, 12, 15, 17] excel at learning how to weigh attributes and link records across data sources. Using machine learning techniques such as decision trees [14] or Baysian Networks [5], they are able to determine which attributes are most relevant to consider when trying to match records across different data sources.

The Apollo system [11, 10] incorporates secondary sources into the record linkage process to improve its performance. It leverages decision tree technology to improve the accuracy of the record linkage process by augmenting primary sources with information obtained from secondary sources. It utilizes an information mediator to determine if there are any available secondary data sources for the given primary sources. If there are one or more available secondary sources, it augments the primary data sources with additional attribute(s). Labeled examples provided by a user allow the system to determine if the newly added attributes are informative enough to incorporate into the mapping rules. With the flexible nature of the record linkage framework used in Apollo, incorporating this additional information from secondary sources is an easy and efficient process. Also, as shown in [10], this approach leads to an improvement in both precision and recall.

Algorithm : ApolloLearner($LE, SS, N, AllCM$)

procedure EvaluateSS(LE, SS)
 $GoodSS \leftarrow \phi$
 for each $src \in SS$
  do {
   $\#true \leftarrow 0$
   $\#false \leftarrow 0$
   for each $example \in LE$
    do {
     $label \leftarrow$ Label($src, example$)
     if $label$
      then $\#true \leftarrow \#true + 1$
      else $\#false \leftarrow \#false + 1$
   if $(\#true \div (\#true + \#false)) > 0.75$
    then $GoodSS \leftarrow GoodSS \bigcup src$
 return ($GoodSS$)

procedure Learn($GoodSS, LE, N, AllCM$)
 $nLEsets \leftarrow$ DivideNSets(LE, N)
 for each $set \in nLEsets$
  do {
   $dt \leftarrow$ LearnDT(set)
   $labels[set] \leftarrow$ Classify($dt, AllCM$)
 $nextExp \leftarrow$ GetInformativeExp($labels$)
 if $nextExp$ **not** ϕ
  then {
   for each $exp \in nextExp$
    do $LE \leftarrow LE \bigcup$ Label($GoodSS, exp$)
   Learn($GoodSS, LE, N, AllCM$)
 return ($labels$)

main
 $GoodSS \leftarrow$ EvaluateSS(LE, SS)
 Learn($GoodSS, LE, N, AllCM$)

Figure 2: Apollo's Unsupervised Learning Algorithm

4 Utilizing Secondary Sources For Unsupervised Learning

In this section we describe Apollo's approach to identifying secondary sources that can be used to accurately classify record pairs as matches or non-matches. Moreover, we present how Apollo utilizes the identified secondary sources in an unsupervised active learning process. Apollo's learning algorithm for using secondary data sources is presented in Figure 2. There are two main procedures in the algorithm: (1) Evaluating secondary sources, and (2) Automatically labeling candidate record pairs.

4.1 Evaluating Secondary Sources

There exist a wide variety of potentially useful secondary sources. While most secondary sources provide pertinent information, not all of them can be used to identify matched records. For example, a company may have multiple locations, therefore if two company records have different location attribute values, this does not imply that the records do not refer to the same company.

In Apollo, we provide a simple mechanism to evaluate the capabilities of various secondary sources with respect to labeling matched records. As shown in the *EvaluateSS* procedure in Figure 2, we begin with a very small set of training data (e.g. 25 record pairs) and a set of available secondary sources. Next, we use each secondary source to label all the record pairs in the training set. Based on the user provided labels and the labels given by the secondary sources, we calculate the percentage of record pairs that are labeled correctly by the secondary source. If the secondary source can classify more then 75%[1] of the record pairs correctly, we classify it as being useful for labeling.

4.2 Unsupervised Active Learning

Once Apollo has identified which secondary source can be used to automatically label record pairs, it can use this secondary source to generate a set of labeled training examples used by the record linkage process.

Apollo utilizes an active learning approach described by Tejada et. al [17] to reduce the number of examples labeled by the secondary source. As shown in the *Learn* procedure in Figure 2, it begins unsupervised active learning by composing a set of labeled examples (LE) using the selected secondary source (source in $GoodSS$) from the matches generated by the candidate generator. It uses a committee of N decision tree learners to identify the most informative examples that should be labeled next. This is done by dividing the labeled example set into N unique sets, and learning a decision tree based on each set. All the candidate matches ($AllCM$) are then classified using the N learned decision trees. The most informative examples are determined by choosing examples from the candidate set with the highest level of disagreement between the N decision trees.

The selected examples are then labeled using information retrieved from the chosen secondary source. In a traditional active learning process, this labeling requires a user or a domain expert. This requirement is negated in Apollo by utilizing secondary sources in place of a user. Once all the selected examples are labeled, Apollo re-learns N decision trees by adding the newly labeled examples to the set of labeled examples. This is done to obtain the next set of the most informative examples. The process is repeated until all decision trees converge.

It should be noted that Apollo is not limited to using one secondary source for labeling examples. If there exist multiple useful secondary sources, Apollo can utilize all these sources to label the informative examples. This is further explored in section 7.

[1]This is a manually selected value and we are working on a method to learn this value automatically.

5 Experimental Evaluation

We evaluated the idea of utilizing secondary sources to label record pairs as matches or non-matches by performing four sets of experiments. The goal of the experiments was to show that we can achieve almost optimal performance from Apollo when performing unsupervised learning using secondary sources for labeling. In all sets of experiments we performed linkage across datasets in the real world restaurant domain.

We used wrapper technology discussed in [13] to extract restaurant records from the Zagat's and Dinesite web sources. Each web source provided a restaurant's name, address, city, state, phone number, and cuisine type. The Zagat data source contained 897 records, while the Dinesite data source contained 1257 records. There were 136 matching records in the two datasets. Due to the inconsistencies between the two sources, a record linkage system was required to find common restaurants. Available secondary data sources included: a geocoder that provided geographic coordinates for a given address, and a postal web site which provided 9-digit zipcode for a given address. Each set of experiments contained 10 runs and the results shown are the average values for all runs.

In the first set of experiments we utilized the geocoder to generate training examples. If the geocoder returned the same geographic coordinates for the address of the two records in a record pair, the record pair was labeled as a match. In the subsequent sets, we performed the experiments using the postal service web site, a random method, and user involvement to label record pairs. In the random method, we randomly generated labels for the training examples. The performance of the Apollo system with each secondary source was compared to the performance of the Apollo system requiring a user to label record pairs and to random labeling. The results were measured using the F-measure formula, which combines precision and recall measures as shown below.

$$F - measure = \frac{2 \times recall \times precision}{recall + precision}$$

Figure 3 shows that the Apollo system with user labeling performs just a little better when compared to the Apollo system with secondary source labeling. In the case of both secondary sources, Apollo performs better then the "strawman" approach of returning a random label when the system asks the user to label a match.

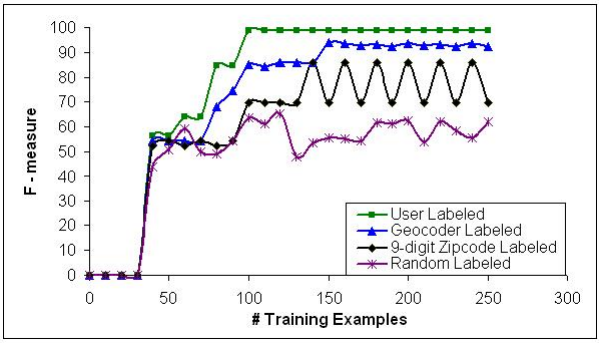

Figure 3: Unsupervised Learning Using Secondary Sources

In particular, the Apollo system with the geocoder secondary source reaches an optimal F-measure of 93.88 with 150 labeled examples. The Apollo system with user labeling reaches an optimal F-measure of 97.22 with 100 labeled examples. However, in the case of the Apollo system with the user labeling, the user has the task of labeling 100 record pairs, while in the Apollo system with automatic unsupervised labeling, the user does not need to label any of these pairs.

The Apollo system with the postal service web site also gets to a F-measure value of 85.71 with 140 labeled examples. Due to the fact that there exist different restaurants with the same 9-digit zipcode, the system alternates between learning strict and looser mapping rules. Due to inaccurate labeling, an oscillation between 100% precision and recall occurs. This oscillating effect can be seen in the zipcode results in Figure 3.

While both secondary sources perform well, they perform worse than the Apollo system with user labeled examples. The key reason behind this is due to the inaccurate labeling mentioned earlier. In general, it would be difficult to find one "golden" secondary source that can accurately identify records as matches or non-matches. We can improve the accuracy of the labeling process by combining information from multiple secondary sources, or by combining secondary source information with attributes of the primary sources to label record pairs.

6 Related Work

There has been significant work done on solving the record linkage problem [1, 2, 3, 6, 8, 12, 15, 17]. This work includes research on entity matching [3, 8], object consolidation [8, 17], and de-duplication [1, 6, 12, 15]. All these systems utilize some form of textual similarity measures to determine if two records should be linked. However, none of the systems incorporate the idea of utilizing secondary sources to obtain relevant information and use this information to improve the record linkage process.

Doan et. al. [3] describe a profiler-based approach to improving entity matching. The key idea in the paper is to design profilers by mining large amounts of data from different web sources, obtaining input from domain experts, or by examining previously matched entities. The profilers generate rules that determine relationships between various attributes of entities, for example someone with age 9 is not likely to have a salary of $200,000. This idea is complementary to our approach of utilizing secondary sources to provide additional attributes.

To the best of our knowledge, our approach is the first to perform unsupervised active learning for record linkage. Most approaches that use learning in the record linkage process require a human in the loop.

7 Discussion

In this article, we presented our approach to utilizing secondary sources for unsupervised record linkage. We showed that a secondary data source can be used to provide training data automatically and free the user from the burden of labeling data. Needless to say that our approach is only applicable in scenarios where there exists viable secondary data sources. However, as we pointed out earilier, for most data sources on the Internet, there exist some secondary sources with relevant information. In some enterprise settings, where one often needs to work with obscure numbers, it may be harder to find secondary data sources.

Our experimental evaluation shows that by utilizing secondary sources to automatically provide training data to record linkage process, Apollo can dramatically reduce a user's involvement while maintaining the accuracy of the record linkage process. In the future, we plan to investigate how Apollo can improve the usage of secondary sources by utilizing combined weights of different secondary data sources and various fields. For example, we can get better labels if we utilize a geocoder in conjunction with cuisine type or restaurant name. Finally, even though the transformations used in Apollo are quite comprehensive, they do not cover all possible sets of transformations. To address this problem, we are working on improving the field (attribute) level matching process. This work applies specific sets of transformations depending on the semantic types of different attributes and leads to more accurate confidence measures for the given attributes.

8 Acknowledgements

This material is based upon work supported in part by the National Science Foundation under Award No. IIS-0324955, in part by the Defense Advanced Research Projects Agency (DARPA), through the Department of the Interior, NBC, Acquisition Services Division, under Contract No. NBCHD030010, in part by the Air Force Office of Scientific Research un-

der grant numbers F49620-01-1-0053 and FA9550-04-1-0105, in part by the United States Air Force under contract number F49620-02-C-0103, and in part by a gift from the Microsoft Corporation.

The U.S. Government is authorized to reproduce and distribute reports for Governmental purposes notwithstanding any copyright annotation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of any of the above organizations or any person connected with them.

References

[1] M. Bilenko and R. J. Mooney. Adaptive duplicate detection using learnable string similarity measures. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-2003)*, pages 39–48, Washington DC, 2003.

[2] S. Chaudhuri, K. Ganjam, V. Ganti, and R. Motwani. Robust and efficient fuzzy match for online data cleaning. In *Proceedings of the 2003 ACM SIGMOD International Conference on Management of Data*, San Diego, CA, 2003. ACM Press.

[3] A. Doan, Y. Lu, Y. Lee, and J. Han. Object matching for data integration: A profile-based approach. In *Proceedings of the IJCAI-03 Workshop on Information Integration on the Web*, 2003.

[4] M. R. Genesereth, A. M. Keller, and O. M. Duschka. Infomaster: An information integration system. In *In Proceedings of ACM SIGMOD-97*, 1997.

[5] D. Heckerman and R. Shachter. Decision-theoretic foundations for causal reasoning. *Journal of Artificial Intelligence Research*, 3:405–430, 1995.

[6] M. A. Hernndez and S. J. Stolfo. The merge/purge problem for large databases. In *Proceedings of the ACM SIGMOD Conference*, 1995.

[7] C.-N. Hsu and M.-T. Dung. Generating finite-state transducers for semistructured data extraction from the web. *Information Systems Journal Special Issue on Semistructured Data.*, 23(8), 1998.

[8] L. Jin, C. Li, and S. Mehrotra. Efficient record linkage in large data sets. In *In Proceedings of the 8th International Conference on Database Systems for Advanced Applications (DASFAA 2003)*, Kyoto, Japan, 2003.

[9] C. Knoblock, S. Minton, J. Ambite, N. Ashish, I. Muslea, A. Philpot, and S. Tejada. The ariadne approach to web-based information integration. *International Journal on Intelligent Cooperative Information Systems (IJCIS)*, 10(1-2):145–169, 2001.

[10] M. Michalowski, C. A. Knoblock, and S. Thakkar. Automatically utilizing secondary sources to align information across data sources. *Submitted to the AI Magazine, Special Issue on Semantic Integration*, 2004.

[11] M. Michalowski, S. Thakkar, and C. A. Knoblock. Exploiting secondary sources for automatic object consolidation. In *Proceeding of 2003 KDD Workshop on Data Cleaning, Record Linkage, and Object Consolidation*, 2003.

[12] A. E. Monge and C. Elkan. The field matching problem: Algorithms and applications. In *Proceedings of the Second Conference on Knowledge Discovery and Data Mining*, pages 267–270, 1996.

[13] I. Muslea, S. Minton, and C. A. Knoblock. Hierarchical wrapper induction for semistructured information sources. *Autonomous Agents and Multi-Agent Systems*, 4(1/2), 2001.

[14] J. R. Quinlan. Improved use of continuous attributes in c4.5. *Journal of Artficial Intelligence Research*, 4:77–90, 1996.

[15] S. Sarawagi and A. Bhamidipaty. Interactive deduplication using active learning. In *Proceedings of the 8th ACM SIGKDD Conference on Knowldge Discovery and Data Minning*, Edmonton, Alberta, Canada, 2002.

[16] S. Tejada, C. A. Knoblock, and S. Minton. Learning object identification rules for information integration. *Information Systems*, 26(8), 2001.

[17] S. Tejada, C. A. Knoblock, and S. Minton. Learning domain-independent string transformation weights for high accuracy object identification. In *Proceedings of the Eighth ACM SIGKDD International Conference*, Edmonton, Alberta, Canada, 2002.