CHRONOS: Facilitating History Discovery by Linking Temporal Records

Pei Li
University of Milan-Bicocca
pei.li@disco.unimib.it

Haidong Wang
Nankai University
alphardwang@gmail.com

Christina Tziviskou
University of Milan-Bicocca
christina.tziviskou@disco.unimib.it

Xin Luna Dong
AT&T Labs-Research
lunadong@research.att.com

Xiaoguang Liu
Nankai University
liuxguang@nankai.edu.cn

Andrea Maurino
University of Milan-Bicocca
maurino@disco.unimib.it

Divesh Srivastava
AT&T Labs-Research
divesh@research.att.com

ABSTRACT

Many data sets contain *temporal records* over a long period of time; each record is associated with a time stamp and describes some aspects of a real-world entity at that particular time. From such data, users often wish to search for entities in a particular period and understand the history of one entity or all entities in the data set. A major challenge for enabling such search and exploration is to identify records that describe the same real-world entity over a long period of time; however, linking temporal records is hard given that the values that describe an entity can evolve over time (*e.g.*, a person can move from one affiliation to another).

We demonstrate the CHRONOS system which offers users the useful tool for finding real-world entities over time and understanding history of entities in the bibliography domain. The core of CHRONOS is a temporal record-linkage algorithm, which is tolerant to value evolution over time. Our algorithm can obtain an F-measure of over 0.9 in linking author records and fix errors made by *DBLP*. We show how CHRONOS allows users to explore the history of authors, and how it helps users understand our linkage results by comparing our results with those of existing systems, highlighting differences in the results, explaining our decisions to users, and answering "what-if" questions.

1. INTRODUCTION

Many data sets contain *temporal records* over a long period of time; each record is associated with a time stamp and describes some aspects of a real-world entity at that particular time. From such data, users often wish to search for entities in a particular period, and understand the history of one entity or all entities in the data set. For example, *DBLP*[1] lists research papers over many decades; *DBLP* users may wish to find authors by name and year, find the publication history and affiliation history of an author, find the number of her co-authors in each year over time, find her research topics over time, and so on.

A major challenge for enabling such search and exploration is to identify records that describe the same real-world entity over a long period of time; only with such an integrated view, we will be able to trace the history of that entity and collect statistics over time. However, linking temporal records is by no means easy. First, we need to be able to link together records for the same real-world entity but at different times. This is hard because entities can evolve over time; for example, a researcher can move from one affiliation to another, change her research topic, and collaborate with different co-authors over time. Thus, records that describe the same real-world entity at different times can contain different values; blindly requiring value consistency of the linked records may cause false negatives. Second, we need to be able to distinguish records that share common attribute values but refer to different real-world entities. This is especially hard for temporal records because it is more likely to find highly similar entities over a long time period than at the same time; for example, having two persons with highly similar names in the same university over the past 30 years is more likely than at the same time. Thus, records that describe different entities at different times can share common values; blindly matching records that have similar attribute values can cause false positives.

We demonstrate the CHRONOS system[2], which offers users a useful tool for finding real-world entities over time and understanding history of entities in the bibliography domain. The core of CHRONOS is a temporal record-linkage algorithm, which is tolerant to value evolution over time [5]. Our algorithm can obtain an F-measure of over 0.9 in linking author records and can fix errors made by *DBLP*. There are two key ideas for the linkage techniques: first, we apply *time decay* that captures the effect of time elapse on entity value evolution; second, we apply *temporal clustering* that considers records in time order and accumulates evidence over time to enable decision making with a global view.

The demonstration illustrates the novel features of CHRONOS and focuses on the following two aspects. First, we show how CHRONOS allows users to explore the history, including searching authors in a particular time period or in a particular affiliation, tracing the history of publications, co-authors, and affiliations of a particular author, and understanding the statistics of authors, pub-

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

Proceedings of the VLDB Endowment, Vol. 5, No. 12

[2] *Chronos* is a Greek God for *time*; he has three heads, a man, a bull, and a lion, showing the importance of "linkage".

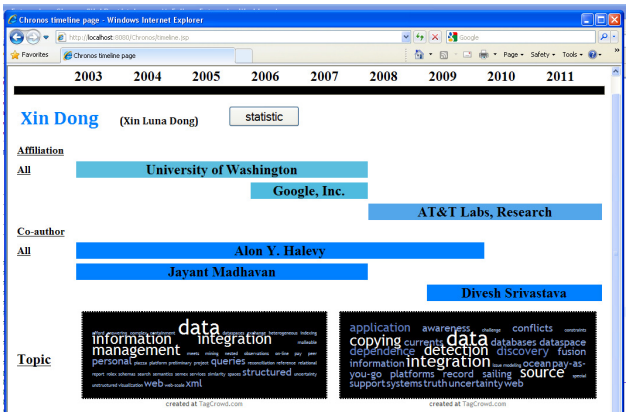

Figure 1: Research history of author "Xin Dong".

lications, etc., over time. Second, we further show how CHRONOS helps users understand our linkage results (linking citation records for the same real-world author) by comparing our results with those of existing systems, such as the manual linkage results from *DBLP*, highlighting differences in the results, explaining to users our decisions, and answering "what-if" questions such as "What would the results look like if we had not applied time decay?" and "What if we had removed these three records?"

In the rest of the proposal, we first describe the features of the CHRONOS system in Section 2, and then describe the system architecture and underlying temporal linkage algorithm in Section 3. We discuss related work in Section 4, and conclude in Section 5.

2. SYSTEM FEATURES

We start by describing the features of CHRONOS through user scenarios. CHRONOS includes all papers collected by *DBLP* till June 1st, 2012. For each paper, we extract a record for each author of that paper, with information for name, paper title, co-authors, conference, and year. In addition, we enrich each author record with information on email and affiliation for the associated time stamp whenever possible and collect such information from digital libraries such as *ACM*[3], *IEEE*[4], *Scopus*[5], journal websites, the PDFs of the papers, and so on.

First, CHRONOS allows users to search for authors over time and find the history of particular authors.

Scenario 1 *Consider a user who would like to find an author named "Xin Dong". She searches "Xin Dong" and* CHRONOS *returns 6 "Xin Dong" entities and 1 "Dong Xin" entity, each one showing the publication period and current affiliation. The user can select one of them, or refine the query by searching "Xin Dong 2011" (the authors named "Xin Dong" and published in 2011) or searching "Xin Dong AT&T" (the authors named "Xin Dong" and was at AT&T at some time").*

Suppose the user has selected one "Xin Dong" entity. She can click the "History" button to trace the history of various aspects of this author, such as her affiliations, co-authors, research topics, and so on. Figure 1 shows a screenshot for this. It shows that this author stayed at "University of Washington" in 2003-2007, at "Google, Inc" in 2006-2007, and at "AT&T Labs" from 2008 till now. Note that this history is purely derived from the author's publications, so may not be precise (e.g., *the author may have joined AT&T Labs in 2007 but started publishing with that affiliation only since 2008). The topic is generated for every five years as the tag cloud*[6] *of publication titles and available abstracts.*

The user can also click the "Statistics" button to see statistics about the author, including graphs of the number of publications by that author over years, the number of co-authors over years, and so on. The user can even see statistics of all authors over years, such as the number of authors and the number of publications. □

Second, in case the user is interested in the author-linkage results, CHRONOS compares its own temporal-linkage results with (1) the manual linkage results by *DBLP*, and (2) the linkage results by BASIC, a traditional record-linkage technique that compares each pair of author records and applies transitivity in clustering the records into author entities [3].

Scenario 2 *Suppose the user is interested in comparing the listed papers by* CHRONOS *and by* DBLP, *she can click the "Comparison" button.* CHRONOS *will show side-by-side the list of papers according to the linkage results by* CHRONOS, *by* DBLP, *and by* BASIC *(see Figure 2).* CHRONOS *also highlights differences between the lists: for each list from* DBLP *and* BASIC, *it highlights the publications not included in its own list; for its own list, it highlights the publications not included in the list from* DBLP *or from* BASIC *(using different colors).*

If the user would like to understand the different decisions, she can click on one highlighted publication and CHRONOS *would explain the reason. For example, if she wonders why publication #22 from* DBLP *is excluded from the list by* CHRONOS, *she can click on #22 and* CHRONOS *would explain "The author 'Xin Dong' of that paper is from 'University of Nebraska-Lincoln', it is unlikely that she moved from 'AT&T Labs' to 'University of Nebraska-Lincoln' in 2010 and moved back to 'AT&T Labs' in 2011". As another example, if she wonders why publication #15 (excluded by* BASIC*) is included in the list by* CHRONOS, *she can click on publication #15 and* CHRONOS *would explain "The author 'Xin Dong' of that paper is from 'University of Washington'; later she moved to 'AT&T Labs' in 2008".*

If the user is curious and would like to understand more, such as why it is considered unlikely for the author to "move from 'AT&T Labs' to 'University of Nebraska-Lincoln' in 2010 and move back to 'AT&T Labs' in 2011" but likely for the author to "move from 'University of Washington' to 'AT&T Labs' in 2008", she can ask for more details. For the previous explanation for publication #22, if the user clicks the "Details" button, the explanation will be extended as "The author was at 'AT&T Labs' in 2008-2010; the probability that she moved to another affiliation in 2010 is .26 and the probability that she moved again in 2011 is .13". Similarly, extended explanation for publication #15 can be "The author was at 'University of Washington' in 2003-2007; the probability that she moved to another affiliation in 2008 is .55". □

Third, for advanced users, CHRONOS answers "what-if" questions and help users compare different results when revising some of the data or applying different linkage methods. Specifically, CHRONOS allows the user to (1) select a subset of records by searching or by selecting on record basis, (2) change the time stamp of some of the selected records, (3) choose to consider decay or not consider decay, and choose to apply different clustering methods, and then compare the results.

Scenario 3 *Consider an advanced user who would like to understand the linkage results further. In particular, she wonders what*

[3]http://dl.acm.org/
[4]http://ieeexplore.ieee.org/
[5]http://www.scopus.com/
[6]http://www.tagcrowd.com/.

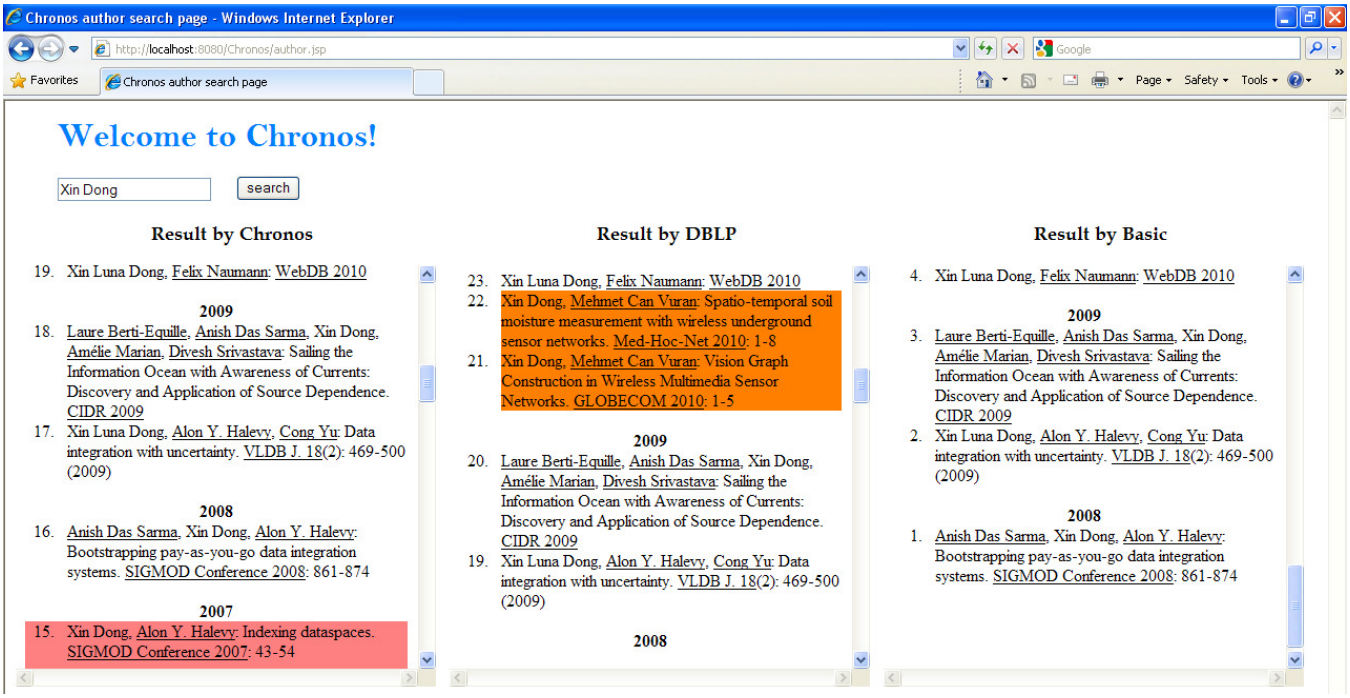

Figure 2: Comparison on publications by "Xin Dong". Only a subset of papers are shown to fit the differences in one screen.

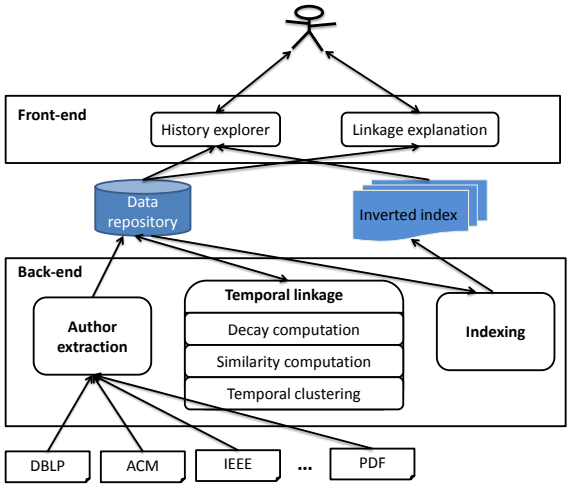

Figure 3: Architecture of the CHRONOS system.

if the two publications with Google Inc. *affiliation were published in 1986-1987 instead of 2006-2007. She could choose all publications by the selected "Xin Dong" entity, and then change the time stamp of the two publications.* CHRONOS *then applies linkage at runtime, shows the publication list from the new results and from the original results side-by-side, and highlights the differences. The difference might be that the two publications are considered to belong to another "Xin Dong" because of the big time gap between the two revised records and the rest of the records.* □

3. FRAMEWORK AND ALGORITHMS

We next describe the architecture of the system and also the key techniques for linking temporal records.

3.1 Architecture

Figure 3 depicts the architecture of CHRONOS. At the back end CHRONOS contains three components in charge of data collection and cleaning: **Author extraction, Temporal linkage** and **Indexing**. At the front end CHRONOS contains two components in charge of interaction with users, search, and decision explanation: **History explorer**, and **Linkage explanation**. We next describe each component in more detail.

Author extraction: This component takes the *DBLP* data as input. For each paper, it extracts records about authors, including author name, paper title, conference, co-author, publication year, and so on. It then follows the links provided by *DBLP* to external sources (*e.g.*, *ACM, IEEE, Scopus*, journal websites, and PDF paper files) and enriches the records by extracting information on affiliation and email of the author at the time of publication. It stores the results in the **Data repository**, which hosts a database using *MySQL*[7].

Temporal linkage: This component identifies author records that refer to the same real-world person. We describe this component in more detail shortly. Note that linkage on the full data set can take hours, but this is performed offline and the results are also stored in **Data repository**, ready for online search.

Indexing: This component builds an **Inverted index** for each identified real-world author. To facilitate search by name, affiliation, and time period, each author is indexed by her names and affiliations over time, and also the years of her publications. We used *Lucene*[8] for indexing.

History explorer: This component is the interface through which the user interacts with the system. It offers (1) author search by name, time period, and affiliation, (2) history tracing for each author, and (3) statistics view of the data. Upon receiving an author query, it finds relevant authors through the **Inverted index**, and then retrieves details about the author from the **Data repository**.

Linkage explanation: This component explains linkage decisions and is in charge of three tasks. First, it shows the comparison of results from CHRONOS, from *DBLP*, and from BASIC. For each selected author, it chooses the cluster from *DBLP* or BASIC with the

[7]http://www.mysql.com.
[8]http://lucene.apache.org.

largest number of publications in the author publication list generated by CHRONOS. Second, it explains the decision of a particular paper included in or excluded from the list of papers for a particular author. Explanations are generated mainly according to the decay, as we define shortly. Third, it performs online temporal linkage and answers "what-if" questions. Note that since online linkage will be performed only on a small subset of records, it is quite efficient and takes only a few seconds.

3.2 Temporal Linkage

The core of the system is the **Temporal linkage** component, which links author records that refer to the same real-world entity. It contains three sub-components: **Decay computation**, **Similarity computation**, and **Temporal clustering**. We next briefly describe the techniques applied in each sub-component, and refer the interested readers to [5] for details.

Decay computation: One key idea of our temporal linkage algorithm is to apply time decay, which aims to capture the effect of time elapse on entity value evolution. Specifically, we define *disagreement decay* as the probability that an entity changes its value of a particular attribute within a particular period of time. Symmetrically, we define *agreement decay* as the probability that two entities share a common value of a particular attribute within a particular period of time. For example, a disagreement decay of .6 for affiliation and 5 years means that the probability that an author changes her affiliation within 5 years is .6; an agreement of .1 for affiliation and 5 years means that the probability that two different authors share the same affiliation within 5 years is only .1. With the use of decay, we do not penalize variety of values over a long time too much, and meanwhile do not reward similarity of values over a long time too much. We learn decay for each attribute from a set of labeled data, for which we know if two records refer to the same entity and if two strings represent the same value.

Similarity computation: We compare a record with a cluster of records considering the following two aspects. First, we consider *value consistency*. We compare the record with the cluster on each attribute, and then take a linear combination of the similarities. In this computation we apply decay, so we are more tolerant to value variety over time. For example, the record of "Xin Dong" from "AT&T" in 2008 has high value consistency with the cluster of "Xin Dong" records from "University of Washington" in 2003-2007, despite the affiliation difference. Second, we consider *continuity*. We compare the time stamp of the record with the time period of the cluster. The higher the continuity, the more likely that the record belongs to the cluster. Our previous example also observes a high continuity, but another record of "Xin Dong" from "RPI" in 1991 has a low continuity with that cluster. The final similarity combines value consistency and continuity.

Temporal clustering: Another key idea of our temporal linkage algorithm is record clustering with a global view of the data. We consider author records in time order and accumulate evidence over time to enable global decision making. Our clustering algorithm proceeds iteratively. In each round, it computes the probability that a record belongs to each cluster according to the record-cluster similarity, and chooses the clustering with the highest probability. It then refines the results iteratively until the results converge.

4. RELATED WORK

There have been several applications for exploring temporal information. BIBNETMINER [8] is the closest to CHRONOS: it collects data from *DBLP* and allows users to explore the history of authors on a time line. However, it takes the linkage results from *DBLP* directly while we focus on enriching and improving DBLP data by applying temporal linkage. INZEIT [7] collects data from New York Times Annotated Corpus and focuses on determining insightful time points as milestones for user queries. PRIMA [6] considers historical data with evolving schemas. None of the systems emphasizes linkage of temporal records.

There have been other systems related to bibliography data. DBLIFE [2] is able to track entities over time, but it applies handcrafted information extraction rules and the entity resolution methods rely heavily on domain knowledge. WINACS [9] extracts data from Web-based information network to perform entity resolution and disambiguation. Again, none of them applies temporal linkage techniques.

Finally, there have been a few demonstrations on record linkage. LINKDB [4] demonstrates the performance and visualization of probabilistic record linkage techniques. SEMEX [1] performs reference reconciliation on personal data. We differ in that we consider linkage of *temporal* records.

5. CONCLUSIONS

This demonstration aims to exhibit the strength of temporal information in information search and exploration. CHRONOS allows users to search for authors by their name, affiliation, and time period; it also allows users to trace the history and statistics of various aspects of an author, a conference, all publications, and so on. The core of the system is a temporal linkage algorithm that is tolerant to value variety over time when identifying records that refer to the same real-world entity. CHRONOS helps users understand its linkage results by comparison with results of other methods, explanation of the differences, and answering "what-if" questions.

6. REFERENCES

[1] Y. Cai, X. L. Dong, A. Halevy, J. M. Liu, and J. Madhavan. Personal information management with SEMEX. In *SIGMOD*, pages 921–923, 2005.

[2] P. DeRose, W. Shen, F. Chen, Y. Lee, D. Burdick, A. Doan, and R. Ramakrishnan. DBLife: A community information management platform for the database research community. In *CIDR*, pages 169–172, 2007.

[3] O. Hassanzadeh, F. Chiang, H. C. Lee, and R. J. Miller. Framework for evaluating clustering algorithms in duplicate detection. *PVLDB*, 2(1):1282–1293, 2009.

[4] E. Ioannou, W. Nejdl, C. Niederée, and Y. Velegrakis. LinkDB: a probabilistic linkage database system. In *SIGMOD*, pages 1307–1310, 2011.

[5] P. Li, X. L. Dong, A. Maurino, and D. Srivastava. Linking temporal records. *PVLDB*, 4(11):956–967, 2011.

[6] H. J. Moon, C. Curino, M. Ham, and C. Zaniolo. PRIMA: archiving and querying historical data with evolving schemas. In *SIGMOD*, pages 1019–1022, 2009.

[7] V. Setty, S. Bedathur, K. Berberich, and G. Weikum. InZeit: efficiently identifying insightful time points. *PVLDB*, 3(2):1605–1608, 2010.

[8] Y. Sun, T. Wu, Z. Yin, H. Cheng, J. Han, X. Yin, and P. Zhao. BibNetMiner: mining bibliographic information networks. In *SIGMOD*, pages 1341–1344, 2008.

[9] T. Weninger, M. Danilevsky, F. Fumarola, J. Hailpern, J. Han, T. J. Johnston, S. Kallumadi, H. Kim, Z. Li, D. McCloskey, Y. Sun, N. E. TeGrotenhuis, C. Wang, and X. Yu. WINACS: construction and analysis of web-based computer science information networks. In *SIGMOD*, pages 1255–1258, 2011.