A Survey on Entity Resolution in Relational Databases

Authors: Gao Guangshang, Zhang Zhixiong

Date: 2016-06-01T00:00:00+00:00

Abstract

[目的]To analyze the research status and future research directions of entity resolution technology in relational databases. [方法]Systematic research is conducted from two aspects of entity resolution: accuracy and efficiency. The accuracy aspect is based on incremental methods, statistical approaches, and relevant information; the efficiency aspect is based on blocking, string similarity, and other techniques. [结果]Maximizing entity resolution accuracy and efficiency constitutes the primary objective of entity resolution technology research; however, significant challenges persist regarding dynamic evolution of data sources, heterogeneity, and approximate string matching. [局限]The discussion is limited to the accuracy and efficiency requirements within the entity resolution process, with insufficient attention devoted to the characteristics and inherent limitations of the resolution models themselves. [结论]This study facilitates a more comprehensive understanding of the entity resolution process, its current research status, and future research directions in relational databases.

Full Text

Preamble

Survey on Entity Resolution over Relational Databases

Gao Guang-shang[1],[2], Zhang Zhi-xiong[1]
[1] National Science Library, Chinese Academy of Sciences, Beijing 100190, China
[2] University of Chinese Academy of Sciences, Beijing 100190, China

Abstract

[**Objective**] To analyze the current status and future research directions of Entity Resolution (ER) technology over relational databases. [**Methods**] Systematic research was conducted on two aspects: accuracy and efficiency of ER. The accuracy of ER was based on incremental methods, statistical methods,

and related information. The efficiency of ER was based on blocking, string similarity, and other techniques. [**Results**] Maximizing precision and efficiency is the main objective of entity resolution, but research on dynamic evolution and heterogeneity of data sources and inexact string matching still faces significant challenges. [**Limitations**] Only precision and efficiency needed in the process of entity resolution are discussed, but the characteristics and limitations of ER models need more attention. [**Conclusions**] This paper will facilitate a comprehensive overview of the process of ER over relational databases, research status, and future research directions.

Keywords: Entity resolution; record linkage; relational databases

Introduction

Research on Entity Resolution (ER) has a long history, with some early work dating back to the 1950s [**?**], yet it remains an active research area today. In 1969, Fellegi and Sunter proposed a technique for linking records based on the assumption that different records referring to the same real-world entity should share certain commonalities [**?**]. This foundational assumption has guided subsequent research in the database community. Entity resolution has been extensively studied under various names, including Record Linkage [**?**], Merge/Purge [**?**], Deduplication [**?**], Reference Reconciliation [**?**], Object Identification [**?**], and others [**?**]. In relational databases, ER primarily aims to identify n ($n > 1$) similar duplicate records that describe the same entity, where records are also referred to as data. The resolution models can be broadly categorized into three types: matching-based clustering (record clustering based on Boolean rule matching information) [**?**, **?**], distance-based clustering (record clustering based on relative distance) [**?**, **?**], and pairwise entity resolution (Pairs ER, which resolves records pair by pair) [**?**, **?**, **?**].

With the advent of the "big data" era, ER technology plays a crucial role in data cleaning, data integration, and data mining, and is therefore considered an important enabling technology for data quality and information sharing. Since ER applications span multiple domains—including census [**?**], public health, web search, product list comparison, counter-terrorism, spam detection, and machine reading—it has attracted attention from experts in both academia and industry. Although ER technology for extraction, matching, and resolution in structured databases is relatively mature, its most challenging problems remain efficiency and accuracy, particularly in complex big data scenarios.

Existing research has introduced and elaborated on the methods involved in each step of the entity resolution process [**?**, **?**], but has not explored resolution strategies from the perspective of ER objectives. Therefore, this survey analyzes and organizes relevant literature from the two aspects of precision and efficiency involved in the ER process, providing an overview of ER research in relational databases. We hope this survey will offer valuable insights and references for future optimization, integration, and further exploration of entity resolution

research.

2 Entity Resolution Technology Overview

To provide a concise yet complete understanding of ER technology in relational databases, this section overviews the relationship between entities and records and the entity resolution process, forming a relatively complete theoretical framework.

2.1 Relationship Between Entities and Records

The described entity may be a physical object (such as a person or a house) or a logical structure (such as a family, a social network, or a list of people who like a particular type of music). These entities are considered to belong to a category set, such as a set of individuals under the person category. Relational data tables are digital representations of these sets. A relational data table contains a series of records or entries, where each record is associated with one or more real-world entities. Specifically, each record may point to a particular entity, but each entity may have one or more records describing it, as shown in [Figure 1: see original paper]. Records consist of columns (attributes, domains). Clearly, all records share the same schema structure, which facilitates the application of entity resolution algorithms.

[Figure 1: see original paper] illustrates three entities (persons) represented by four records, with two duplicate records (records with NAME attribute value "Tom"). We refer to these as similar duplicate records. Typically, people want to remove similar duplicate records from relational data tables because they are one of the key issues affecting data quality, such as in data integration systems. Reasonable approaches include either merging similar duplicate records into a single record or linking each similar duplicate record. As M LLER H et al. [**?**] point out, the task of resolving these duplicate records to make them point to the same entity is not only extremely difficult but also potentially very meaningful.

2.2 Entity Resolution Process

In relational databases, the entity resolution process is typically divided into three stages: pre-linking, linking, and post-linking [**?**, **?**].

(1) Pre-linking Stage. Records are preprocessed or normalized to improve linking accuracy. This stage is likely the most context-dependent of the three, as its goal is to transform attribute data of records to make linking operations as easy as possible. Potential transformations may include converting data in attributes such as dates, phone numbers, and addresses into standard format representations, or splitting/merging data in collections to match another schema.

(2) Linking Stage. This stage performs the actual record linking, including deterministic linking and probabilistic linking [**?**]. Deterministic linking involves

exact matching of one or more attributes and is further divided into simple deterministic linking and transitive deterministic linking. Simple deterministic linking links records if they have identical values in given corresponding attributes, representing the simplest approach. Transitive deterministic linking links records if there is any attribute value match among them, enabling inference that multiple records point to the same entity even when some attribute values are missing.

Probabilistic linking, also known as fuzzy linking, links records when two attribute values do not match exactly. The most famous is the Fellegi-Sunter model [**?**], which describes a set of rules proven to be optimal under the condition of attribute conditional independence. Its main idea is to calculate the discriminatory power of each attribute and then combine these attributes to obtain a probability that two records refer to the same entity. However, probabilistic linking is not easy to implement in large datasets. Moreover, because probabilistic linking methods consider the probability of matches (records or attributes), they may produce false matches (records incorrectly classified as matches) while reducing false non-matches (records incorrectly classified as non-matches).

(3) Post-linking Stage. This stage reviews linking results, examines "possible links," and ultimately uses these results. Three operations exist: linking, merging, and duplicate record deletion. Linking adds a reference in each record pointing to other linked records or stores these links in different datasets. Merging creates a unified view over two datasets if linking different datasets. Duplicate record deletion retains only one necessary record to avoid redundancy.

3 Accuracy-Focused Entity Resolution

Resolution accuracy primarily concerns the quality of record comparison techniques, which is reflected in the comparison methods employed during the resolution process. Current research mainly includes three categories: incremental methods, statistical methods, and related information-based methods.

3.1 Incremental Methods

Unlike scenarios that assume rules and data remain fixed during resolution, incremental methods treat rules and data as dynamically changing, thus adapting well to big data environments with complex structures and rapid updates.

(1) Rule Evolution. SE Whang et al. [**?**] addressed the problem of mutual influence among ER results and proposed an entity resolution method for evolving resolution rules based on dynamic semantics and relationships between rules. This method considers how to leverage existing resolution results to deeply study the ER problem. Specifically, the authors formalized rule evolution, proposed two constraints—rule monotonicity and context-freeness—and noted that rules satisfying these constraints can be processed incrementally. Since the resolution

process using new rules can utilize previous results, computational complexity is reduced and resolution accuracy is improved.

Steven Euijong Whang et al. [**?**] noted that entity resolution is not a one-time process but evolves as people' s understanding of data, schemas, and applications deepens. In most cases, the logic rules used to resolve records continuously evolve because applications themselves evolve and expertise levels for comparing records continuously improve. By incorporating these changing factors, resolution accuracy can be continuously enhanced. The authors argue that the naive approach of re-resolving from scratch for large-scale datasets is intolerable due to high computational costs.

Steven Euijong Whang et al. [**?**] proposed an incremental entity resolution scheme for evolving rules. By leveraging iterative blocking and joint entity resolution, this scheme provides excellent scalability and accuracy and can be applied to different application domains.

(2) Data Evolution. In practice, data blocking methods cannot guarantee independence between blocks because some similar records may be assigned to different blocks. While blocking improves resolution efficiency, it also reduces accuracy. To address this, Steven Euijong Whang et al. [**?**] proposed an iterative entity resolution method based on incremental computation. In each iteration, the ER results from each block computed in the previous iteration are transmitted to other blocks, and each block incrementally computes its internal ER results based on the received updates. This iterative computation continues until results no longer change or the iteration count reaches a given threshold. This method improves result accuracy while ensuring resolution efficiency.

Anja Gruenheid et al. [**?**] observed that data update speeds in the big data era are often fast, causing previous resolution results to quickly become outdated. To solve this, the authors proposed an end-to-end framework that updates resolution results incrementally when data updates (including insertions, deletions, and modifications) arrive. Importantly, without affecting original results, the proposed algorithm can not only merge/separate records from updates with existing clusters but also use new evidence from data updates to correct previous resolution errors. Experiments show that the algorithm significantly reduces resolution time while maintaining quality.

Sunita Sarawagi et al. [**?**] approached from a different perspective, proposing a method for top-k count queries that "resolves entities while solving queries." The algorithm' s foundation is that general queries involve relatively small numbers of data records, making it unnecessary to run ER algorithms on all records—only those involved in query results need processing. The difficulty lies in that resolving records in query results may require records outside the query results, and quickly obtaining relevant records outside query results is also challenging.

Benjelloun et al. [**?**] proposed the "F-Swoosh" algorithm, which adapts well to incremental data scenarios and considers newly added data or features. Heiko Müller et al. [**?**] noted that data cleaning is a time-consuming and costly task.

After obtaining a clean dataset, when a record value changes, the cleaning process only needs to start from records containing that changed value, avoiding cleaning the entire database. Hernandez et al. [**?**] argued that the time and space required to concatenate all data before merging and cleaning are prohibitively expensive, and thus proposed an incremental algorithm that can effectively resolve newly added data in a short time.

Another closely related technique is incremental graph clustering. Claire Mathieu et al. [**?**] studied incremental correlation clustering, noting that when data sources continuously change and evolve, applying resolution methods from scratch for each update is costly. To address speed issues, the authors adopted incremental clustering techniques, focusing on two points: (1) adding one node at a time, and (2) preserving identified clustering results. Charikar, M. et al. [**?**] studied incremental clustering, which differs from other methods by requiring the number of clusters in the result to be preset. This method' s idea is to minimize the maximum cluster diameter formed by the incremental clustering algorithm given a data stream. The authors defined the incremental clustering problem as: for an update sequence containing n nodes (data), maintain a set of k clusters such that whenever an input node appears, it is either assigned to one of the current clusters or a singleton cluster containing only that node is added to the set.

Furthermore, since entity resolution for data evolution is closely related to clustering data streams, Aggarwal et al. [**?**] proposed the CluStream algorithm. Considering that data streams have continuously time-varying characteristics, the algorithm can perform clustering well over different time intervals in evolving environments.

3.2 Statistical Methods

Related to statistical methods is the feature selection problem, where the quality of feature selection directly determines resolution accuracy. Although statistical methods increase inference and learning complexity, they can effectively improve traditional ER algorithms by leveraging previously neglected data attributes, thereby enhancing resolution accuracy.

Xin Dong et al. [**?**] studied utilizing three main features between records to implement an effective machine learning-based ER algorithm. First, they used relationships between records to design new comparison methods. Then, they propagated decision information (match or non-match) between records to accumulate positive and negative evidence. Finally, they gradually enriched each record' s information by merging attribute values, thereby improving entity resolution accuracy. Parag Singla et al. [**?**] proposed a joint inference method that simultaneously reasons about all candidate matching pairs and allows information to propagate from one candidate pair to another via their shared attributes. Since this method is based on Conditional Random Fields (CRF), it improves entity resolution accuracy.

Moreover, in statistical ER methods, parameter setting errors and missing training data can lead to inaccurate detection results. To address these issues, Peter Christen et al. [**?**] proposed a two-stage statistical method. In the first stage, high-quality training examples are automatically selected from record pairs participating in comparison. In the second stage, these training examples are used to train an SVM classifier. Since this two-stage method can effectively adjust the resolution process, it improves entity resolution accuracy.

Lou Junjie et al. [**?**] introduced a variable-weight rule into the Markov Logic Networks (MLNs)-based ER algorithm system, attempting to solve the record ambiguity problem that the original system could not handle (where "John Smith"appearing in two records does not refer to the same person). By introducing variable-weight rules that better reflect real-world situations, the proposed algorithm can improve resolution accuracy to some extent.

3.3 Related Information-Based Methods

Although traditional ER algorithms typically match records individually using various attribute similarity measures, leveraging other related information to assist the ER process enables algorithms to adapt well to large datasets and provides good scalability and flexibility.

Surajit Chaudhuri et al. [**?**] expanded given reference entity tables by mining document collections and using multiple variant forms of each entity in reference entity tables, thus forming a dictionary of string equivalence relationships. Since the method can use precise information from the dictionary to calculate similarity between entities, it improves entity resolution accuracy. Liangcai Shu et al. [**?**] proposed a generative latent topic model called the LDA-dual model that describes relationships between entities and presented a high-precision ER algorithm. Since this model can use global information in corpora to learn a high-performance classifier, it improves resolution accuracy.

Vibhor Rastogi et al. [**?**] proposed an extended ER algorithm that can use intermediate comparison results for comprehensive reasoning. Since this algorithm not only utilizes similarity information between records and co-occurrence frequency information but also fully considers the influence between record comparison results, it improves entity resolution accuracy.

3.4 Comparative Analysis of Methods

Accuracy-focused entity resolution primarily concerns the quality of comparison techniques between similar duplicate records. A comparison of the main research methods is shown in .

**** Comparison of main accuracy-focused research methods (primarily concerning quality of comparison techniques between similar duplicate records)

Method	Advantages	Disadvantages
Incremental Methods: Rule evolution; Data evolution	Effectively reusing previous resolution results improves both efficiency and accuracy; The clustering algorithm used is an unsupervised learning algorithm that can assist similarity calculation and is well-suited for resolving similar duplicate records; Uses correlation results to iteratively find similar duplicate records; Can achieve very high accuracy.	Lacks certain flexibility; The resolution optimization process requires large time overhead.
Statistical Methods: Conditional random fields; Statistical methods; Markov Logic Networks, etc.	Easy to extend; High efficiency for small datasets, but efficiency often cannot be further improved as data scale expands; Does not depend on specific application domains; Considers hidden relationships between attributes behind vocabulary; Can search for corresponding field matching functions, avoiding accuracy fluctuations caused by applying fixed matching functions to different data sources.	High cost of manual annotation; High computational complexity; Parameters are difficult to determine, prone to overfitting; Parameter determination depends on domain knowledge; Lacks specific standards.

Method	Advantages	Disadvantages
Related Information-Based Methods: Equivalence relation dictionary; Global information in corpora; Similarity information, co-occurrence information	Introduces uncertainty; Uses complex structures; Generally mixed with noise, leading to lower accuracy in resolution results.	Different field similarity calculation methods are often particularly effective for specific string types; Since similarity between attributes and similarity between records is a complex nonlinear relationship, merging all attribute values into one long string or simply using weighted sums of attribute similarities to calculate record similarity is not advisable.

4 Efficiency-Focused Entity Resolution

Resolution efficiency primarily concerns the execution speed of resolution algorithms, reflected in two aspects: reducing the number of record pair comparisons needed and improving comparison efficiency of record attribute values. Current research mainly includes blocking-based methods, string similarity-based methods, and other techniques. Although many researchers have made tremendous efforts to improve resolution efficiency, existing algorithms still have worst-case time complexity of $2()On$ [**?**], meaning computational complexity remains far beyond linear, making it difficult to apply to large data scenarios.

4.1 Blocking-Based Methods

When the scale of records to be compared is large, the basic traditional technique uses a "nested" loop to compare record pairs one by one, requiring substantial computational overhead. The purpose of blocking methods is to reduce the comparison space, thereby decreasing the number of record comparisons and ultimately achieving high resolution efficiency without affecting accuracy and completeness. We review existing research from three aspects: attribute values, automatic learning, and blocking method comparisons.

(1) Attribute Values. To improve ER efficiency, Hernández MA et al. [**?**] early proposed the idea of data blocking processing. First, records are sorted separately according to different attribute values. Then, a fixed-length window sequentially scans each record sequence, and matching operations are performed within the window. Finally, matching results from multiple attributes are merged to obtain the final ER results. Assuming window size is l and record count is n, this method can reduce ER cost from $2()On$ to $(Oln$, significantly improving ER efficiency in practice. However, the worst-case scenario for l is n

while maintaining ER accuracy, so the algorithm' s worst-case time cost remains $2()On$.

Andrew McCallum et al. [**?**] utilized data blocking ideas with a low-cost distance metric to effectively divide data into overlapping subsets. This method first partitions records into independent blocks based on certain attribute values, then runs clustering algorithms separately within each block, and finally merges clustering results from each block to obtain ER results. This method reduces the time cost of each clustering algorithm call and improves the overall efficiency of clustering-based ER algorithms.

Zhen Lingmin et al. [**?**] addressed ER efficiency in relational databases by proposing a method to calculate record attribute weights using information gain and probability statistics based on blocking techniques. These weights represent the importance of current attributes in records. By calculating each attribute' s weight separately to fully reflect the importance of key attributes, which is more realistic, this approach not only improves resolution efficiency but also maintains accuracy.

(2) Automatic Learning. Hung-sik Kim et al. [**?**] proposed an iterative Locality-Sensitive Hashing (LSH) algorithm for large-scale data collections to achieve fast and accurate blocking. Since this algorithm can dynamically merge LSH-based hash tables, it can quickly block data. Importantly, the authors also provided corresponding resolution algorithms with certain advantages in resolution speed, thus improving resolution efficiency.

Rares Vernica et al. [**?**] studied how to effectively execute ER in parallel, proposing to use cloud computing environments (MapReduce) to accelerate ER efficiency on large-scale data. By presenting a three-stage method based on data blocking computation in cloud computing environments, solutions can be explored stage by stage, providing new ideas for efficient ER processes.

Mikhail Bilenko et al. [**?**] introduced an adaptive framework to automatically learn blocking functions that guarantee both efficiency and accuracy. By proposing two learnable blocking function methods based on predicates and providing a learning algorithm to train them, this machine learning-based adaptive data blocking strategy can improve resolution efficiency.

(3) Blocking Method Comparisons. Rohan Baxter et al. [**?**] comprehensively reviewed various data blocking strategies in ER methods, comparing bigram indexing and Canopy clustering methods with standard traditional blocking algorithms and sorted-neighborhood blocking methods. Results show that bigram indexing and Canopy clustering can provide scalable blocking methods with potential for speed improvement and accuracy enhancement.

Toralf Kirsten et al. [**?**] formally described and comparatively analyzed two frequently used data blocking methods in practice. One uses simple strategies (e.g., randomly selected hash functions) to partition data into blocks, while the other uses semantic information (e.g., descriptive rules based on attribute val-

ues). The latter method has obvious advantages in ER time efficiency. However, finding rules with suitable semantic information in practice is very difficult and sometimes even impossible.

4.2 String Similarity-Based Methods

Most applications assume attribute values are strings during comparison. Therefore, exploring differences between two strings at the character level and substring level and designing effective string similarity algorithms are important considerations.

(1) String Features. Nick Koudas et al. [**?**] early proposed optimization for ER based on string similarity. By deploying flexible multi-attribute string matching schemes on large databases, preliminary optimization algorithms could be provided, though these used semantic equivalence information that cannot be captured textually. Chaudhuri, S. et al. [**?**] further abstracted ER problems based on string similarity matching on relational data, proposing "similarity join" and "similarity query" operations as fundamental database operations.

Chuan Xiao et al. [**?**] addressed similarity join problems by transforming string similarity computation into set similarity join problems and proposing a set similarity join operation algorithm. By combining prefix and suffix filtering methods based on strings, the proposed method can use order information to avoid similarity computation for all possible record pairs, thereby improving efficiency of similarity join-based resolution methods.

Panagiotis Papapetrou et al. [**?**] addressed long string similarity queries using precomputed alignment scores for variable-length strings, proposing a variable-length string search method that improves ER efficiency for long string attribute values.

(2) n-gram. Chen Li et al. [**?**] studied approximate string matching based on n-grams. The basic idea is to build n-gram indexes on strings, transform string distances into the number of corresponding n-gram intersections, and then provide efficient similarity join algorithms based on n-gram set semantics, improving ER efficiency. Behm, A. et al. [**?**] addressed the problem of large index space by proposing an inverted index method to accelerate similarity queries. By discarding string lists and combining related lists to reduce index space, this method can maintain effective query processing and improve ER efficiency.

Qiu Yuefeng et al. [**?**] proposed an efficient n-gram-based clustering algorithm that uses a priority queue algorithm to accurately cluster similar duplicate records during clustering. Extensive experimental data demonstrates the rationality and efficiency of this resolution method. Since the algorithm can adapt to common spelling errors such as insertion, deletion, substitution, transposition, and word swapping, it has good resolution efficiency with complexity of only $()On$.

4.3 Other Methods

Effectively considering other important information during the ER process can greatly reduce data processing time and space complexity, thereby improving resolution efficiency. Such information includes graphics processor characteristics, entity evolution over time, data noise in big data environments, human-machine hybrid methods, and big data tools.

Michael D. Lieberman et al. [**?**] studied ER on high-dimensional data, proposing a GPU-based similarity join algorithm called LSS. By using hash technology and combining GPU characteristics to provide efficient implementations for two basic data sorting and retrieval operations, this algorithm is very suitable for similarity join resolution operations on high-dimensional data.

Yan Cairong et al. [**?**] proposed an iterative parallel processing framework based on the MapReduce programming model. Using a learning-oriented classification method for entity resolution and leveraging attribute similarity transitivity combined with functional language characteristics, records are efficiently aggregated. Since the MapReduce programming model is very suitable for integrated ER processing, the proposed parallel framework features fast programming and efficient execution, while data partitioning and parallel processing techniques avoid memory overflow issues caused by large numbers of joins. Yan Cairong et al. [**?**] proposed a machine computation and crowdsourcing combined ER method. This method first uses the MapReduce parallel computing framework to exclude impossible record pairs, thereby reducing the number of human intelligence tasks, followed by deterministic manual annotation. To support privacy protection, a role-based access control model and important information hiding strategy were proposed for crowdsourcing computation. By fully utilizing the advantages of both machine and manual processing, this human-machine combined method can better guarantee both high efficiency and high accuracy in the resolution process while effectively avoiding information leakage.

Wang Ning et al. [**?**] noted that traditional ER algorithms perform poorly in efficiency, quality, and particularly noise resistance in big data environments. They proposed a two-tiered correlation clustering algorithm (Two-Tiered). Based on correlation clustering and introducing neighbor relationships of nodes that can effectively define associations between nodes and classes, the proposed algorithm outperforms traditional algorithms in computational cost, noise resistance, and scalability.

Yang Dan et al. [**?**] studied how to resolve entities with temporal information in data spaces, proposing a four-stage time-centered collective entity resolution strategy (T-CER) based on time-based clustering (T-Clustering). T-CER considers the role of temporal information at different stages of the ER process and uses temporal constraints to check resolution results. By combining data heterogeneity and temporal evolution characteristics, the proposed resolution method is more feasible and effective.

4.4 Comparative Analysis of Methods

Efficiency-focused entity resolution primarily concerns the efficiency of comparison processes between similar duplicate records. A comparison of main research methods is shown in .

**** Comparison of main efficiency-focused research methods (primarily concerning efficiency of comparison processes between similar duplicate records)

Method	Advantages	Disadvantages
Blocking-Based Methods: Attribute values; Automatic learning; Blocking method comparison	Effectively compresses feature attribute dimensions and obtains record representatives within groups, laying a foundation for subsequent efficient and accurate resolution; Greatly reduces the number of comparison computations, thereby reducing computational complexity to some extent; Requires little memory for the resolution process, enabling effective resolution of large numbers of similar duplicate records.	Efficiency largely depends on selected keys; Key selection usually depends on domain knowledge, requiring participation of experts with deep domain understanding, which reduces automation and increases result uncertainty; Inappropriate key selection can cause large amounts of duplicate data to be divided into different subsets, reducing match quantity; May affect resolution result completeness.
String Similarity-Based Methods: String features; n-gram	Uses mature and reliable string algorithms; Handles character spelling errors well; Has good scalability; Effectively solves the complex nonlinear relationship between attribute and record similarity.	Different field similarity calculation methods are often particularly effective for specific string types; Since similarity between attributes and similarity between records is a nonlinear mapping relationship, merging all attribute values into one long string or simply using weighted sums of attribute similarities to calculate record similarity is not advisable.

Method	Advantages	Disadvantages
Other Methods: Graphics processors; Entity temporal evolution characteristics; Big data environment noise; Human-machine hybrid methods; Big data tools	Fully utilizes corresponding characteristics to design better matching functions; Fast computation speed.	These methods have their own strengths, but none is universal for all datasets; Reduces human factor influence; Poor scalability and self-adaptability.

5 Conclusions and Future Directions

Existing research on entity resolution technology in relational databases has primarily focused on precision and efficiency, seeking a suitable compromise strategy between them. Although current research attempts to improve ER technology overall, there is a lack of ER technology suitable for big data environments, particularly regarding dynamic evolution, heterogeneity, and inexact string matching of data sources. This includes entity resolution for time-varying dynamic data, large-scale identity management, privacy and query-driven ER, and active learning and crowdsourcing-based ER. Additionally, although graph-based reasoning and resolution requirements exceed current theoretical applications, they represent a feasible solution. Particularly, incremental and distributed resolution strategies can significantly improve both resolution accuracy and efficiency while providing good scalability and efficiency.

With continuously expanding application scales, rapidly growing data volumes, increasingly complex data relationships, and rising data processing requirements, traditional one-to-one record comparison is often not optimal, as it requires substantial resolution time, making it difficult to meet efficiency requirements and even more challenging for complex big data environments. Therefore, we identify three open research directions for future ER technology:

(1) Dynamic Evolution of Data Records. Complex data records in some applications are frequently updated, such as information on the Internet and social networks. How to perform fast and effective entity resolution on frequently updated dynamic complex datasets is a major challenge for ER technology.

(2) Integration of Data Records. Extracting, cleaning, and integrating heterogeneous and massive data sources is a prerequisite for effectively using such data. This brings challenges of uncertain data, structural inconsistencies, and schema matching between data records. How to accurately resolve multiple data

records describing the same entity under these conditions is a major challenge for ER technology.

(3) Inexact String Matching. Comparison between data records is a computationally expensive process, and since the number of matching record pairs is often far less than non-matching pairs, most comparison processes are wasted on non-matching record pairs. Therefore, researching fundamental methods such as inexact string matching methods and optimal filter selection during character matching to minimize the number of record pairs that need comparison while ensuring matching accuracy and completeness is a major challenge for ER technology.

Author Contributions:
Gao Guang-shang: Research implementation, literature investigation, analysis, paper writing, final version revision.
Zhang Zhi-xiong: Research conceptualization.

Corresponding Author: Gao Guang-shang, E-mail: gaoguangshang@mail.las.ac.cn

6 References

[1] NEWCOMBE H B, KENNEDY J M, AXFORD S, et al. Automatic Linkage of Vital Records [J]. Science, 1959, 130(3381):954-959.
[2] FELLEGI I P, SUNTER A B. A theory for record linkage [J]. Journal of the American Statistical Association, 1969, 64(328):1183-1210.
[3] NEWCOMBE H B, KENNEDY J M. Record linkage: making maximum use of the discriminating power of identifying information [J]. Commun ACM, 1962, 5(11):563-566.
[4] HERNÁNDEZ M A, STOLFO S J. The merge/purge problem for large databases [C] // in: Proceedings of the 1995 ACM SIGMOD international conference on Management of data, San Jose, California, USA. 223807: ACM, 1995: 127-138.
[5] SARAWAGI S, BHAMIDIPATY A. Interactive deduplication using active learning [C] // in: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, Edmonton, Alberta, Canada. 775087: ACM, 2002: 269-278.
[6] DONG X, HALEVY A, MADHAVAN J. Reference reconciliation in complex information spaces [C] // in: Proceedings of the 2005 ACM SIGMOD international conference on Management of data, Baltimore, Maryland. 1066168: ACM, 2005: 85-96.
[7] TEJADA S, KNOBLOCK C A, MINTON S. Learning object identification rules for information integration [J]. Inf Syst, 2001, 26(8):607-633.
[8] PETER C. Data Matching Concepts and Techniques for Record Linkage, Entity Resolution, and Duplicate Detection [M]. springer. 2012.
[9] ELMAGARMID A K, IPEIROTIS P G, VERYKIOS V S. Duplicate Record Detection: A Survey [J]. IEEE Trans on Knowl and Data Eng, 2007, 19(1):1-16.
[10] WINKLER W E. Overview of record linkage and current research directions

[R]. Citeseer: BUREAU OF THE CENSUS, 2006.
[11] BENJELLOUN O, GARCIA-MOLINA H, MENESTRINA D, et al. Swoosh: a generic approach to entity resolution [J]. The VLDB Journal, 2009, 18(1):255-276.
[12] BHATTACHARYA I, GETOOR L. Collective entity resolution in relational data [J]. ACM Trans Knowl Discov Data, 2007, 1(1):5.
[13] MANNING C D, RAGHAVAN P, SCHÜTZE H, et al. Introduction to Information Retrieval [M]. Cambridge University Press. 2008: 496.
[14] ARASU A, GÖTZ M, KAUSHIK R, et al. On active learning of record matching packages [C] // in: Proceedings of the 2010 ACM SIGMOD International Conference on Management of data, Indianapolis, Indiana, USA. 1807252: ACM, 2010: 783-794.
[15] 刘骏豪, 孙晶莹. 2011 年德国人口普查中的新技术——记录连接 [J]. 中国统计, 2011, (11):38-39.
[16] 谭明超, 刁兴春, 曹建军. 实体分辨研究综述 [J]. 计算机科学, 2014, 41(4):9-12, 20. (TAN Ming-chao, DIAO Xing-chun, CAO Jian-jun. Survey on Entity Resolution. Computer Science. 2014, 41(4):9-12, 20.)
[17] MÜLLER H, FREYTAG J-C. Problems, methods, and challenges in comprehensive data cleansing [M]. Professoren des Inst. Für Informatik. 2005.
[18] Record Linkage in Large Data Sets[EB/OL].[2014-12-02]. http://www.danisola.com/record-linkage-in-large-data-sets/.
[19] REITER J. Data Quality and Record Linkage Techniques [J]. Journal of the American Statistical Association, 2008, 103(482):881-881.
[20] WINKLER W E. Methods for record linkage and bayesian networks [R]. Statistical Research Division, US Census Bureau, Washington, DC, 2002.
[21] WHANG S E, GARCIA-MOLINA H. Entity resolution with evolving rules [C] // in: Proceedings of the VLDB Endowment, Singapore, 2010: 1326-1337.
[22] WHANG S E, GARCIA-MOLINA H. Incremental entity resolution on rules and data [J]. The VLDB Journal, 2014, 23(1):77-102.
[23] WHANG S E, GARCIA-MOLINA H. Developments in generic entity resolution [J]. IEEE Data Engineering Bulletin, 2011, 13(11):24-30.
[24] WHANG S E, MENESTRINA D, KOUTRIKA G, et al. Entity resolution with iterative blocking [C] // in: Proceedings of the 2009 ACM SIGMOD International Conference on Management of data, Providence, Rhode Island, USA. 1559870: ACM, 2009: 219-232.
[25] GRUENHEID A, DONG X L, SRIVASTAVA D. Incremental Record Linkage [C] // in: Proceedings of the VLDB Endowment, Hangzhou, China. VLDB Endowment, 2014: 20-12.
[26] SARAWAGI S, DESHPANDE V S, KASLIWAL S. Efficient top-k count queries over imprecise duplicates [C] // in: Proceedings of the 12th International Conference on Extending Database Technology: Advances in Database Technology, Saint Petersburg, Russia. 1516413: ACM, 2009: 450-461.
[27] HERNÁNDEZ M A, STOLFO S J. Real-world Data is Dirty: Data Cleansing and The Merge/Purge Problem [J]. Data Min Knowl Discov, 1998, 2(1):9-37.
[28] MATHIEU C, SANKUR O, SCHUDY W. Online correlation clustering [J]. arXiv preprint arXiv:10010920, 2010, 12(3):21-36.

[29] CHARIKAR M, CHEKURI C, FEDER T, et al. Incremental clustering and dynamic information retrieval [C] // in: Proceedings of the twenty-ninth annual ACM symposium on Theory of computing, ACM, 1997: 626-635.
[30] AGGARWAL C C, HAN J, WANG J, et al. A framework for clustering evolving data streams [C] // in: Proceedings of the 29th international conference on Very large data bases - Volume 29, Berlin, Germany. 1315460: VLDB Endowment, 2003: 81-92.
[31] SINGLA P, DOMINGOS P. Collective object identification [C] // in: Proceedings of the 19th international joint conference on Artificial intelligence, Edinburgh, Scotland. 1642589: Morgan Kaufmann Publishers Inc., 2005: 1636-1637.
[32] CHRISTEN P. Automatic record linkage using seeded nearest neighbour and support vector machine classification [C] // in: Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining, Las Vegas, Nevada, USA. 1401913: ACM, 2008: 151-159.
[33] 楼俊杰, 徐从富, 郝春亮. 基于马尔科夫逻辑网络的实体解析改进算法 [J]. 计算机科学, 2010, (08):243-247. (LOU Jun-jie, XU Cong-fu, HAO Chun-liang. Improvement of Entity Resolution Based on Markov Logic Networks. Computer Science, 2010, (08):243-247.)
[34] CHAUDHURI S, GANTI V, XIN D. Mining document collections to facilitate accurate approximate entity matching [J]. Proc VLDB Endow, 2009, 2(1):395-406.
[35] LIANGCAI S, BO L, WEIYI M. A Latent Topic Model for Complete Entity Resolution [C] // in: Data Engineering, 2009 ICDE '09 IEEE 25th International Conference on, 2009: 880-891.
[36] RASTOGI V, DALVI N, GAROFALAKIS M. Large-scale collective entity matching [C] // in: Proceedings of the 37th International Conference on Very Large Data Bases, Seattle, Washington. USA: VLDB, 2011: 208-218.
[37] GETOOR L, MACHANAVAJJHALA A. Entity resolution: theory, practice & open challenges [J]. Proc VLDB Endow, 2012, 5(12):2018-2019.
[38] MCCALLUM A, NIGAM K, UNGAR L H. Efficient clustering of high-dimensional data sets with application to reference matching [C] // in: Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining, Boston, Massachusetts, USA. USA: ACM, 2000: 169-178.
[39] 甄灵敏, 杨晓春, 王斌, et al. 基于属性权重的实体解析技术 [J]. 计算机研究与发展, 2013, (S1):281-289. (Zhen Lingmin, Yang Xiaochun, Wang Bin, and Ahmed A Hussein. An Entity Resolution Approach Based on Attributes Weights. Journal of Computer Research and Development, 2013, (S1):281-289.)
[40] KIM H-S, LEE D. HARRA: fast iterative hashed record linkage for large-scale data collections [C] // in: Proceedings of the 13th International Conference on Extending Database Technology, Lausanne, Switzerland. USA: ACM, 2010: 525-536.
[41] VERNICA R, CAREY M J, LI C. Efficient parallel set-similarity joins using MapReduce [C] // in: Proceedings of the 2010 ACM SIGMOD International Conference on Management of data, Indianapolis, Indiana, USA. 1807222: ACM, 2010: 495-506.

[42] BILENKO M, KAMATH B, MOONEY R J. Adaptive Blocking: Learning to Scale Up Record Linkage [C] // in: Data Mining, 2006 ICDM ' 06 Sixth International Conference on, USA. USA: IEEE, 2006: 87-96.
[43] BAXTER R, CHRISTEN P, CHURCHES T. A Comparison of Fast Blocking Methods for Record Linkage [C] // in: the First Workshop on Data Cleaning, Record Linkage and Object Consolidation, KDD, Washington, DC. USA: KDD, 2003: 25-27.
[44] KIRSTEN T, KOLB L, HARTUNG M, et al. Data partitioning for parallel entity matching [J]. arXiv preprint arXiv:10065309, 2010, 10(4):20-29.
[45] KOUDAS N, MARATHE A, SRIVASTAVA D. Flexible string matching against large databases in practice [C] // in: Proceedings of the Thirtieth international conference on Very large data bases - Volume 30, Toronto, Canada. 1316782: VLDB Endowment, 2004: 1078-1086.
[46] CHAUDHURI S, GANTI V, KAUSHIK R. A Primitive Operator for Similarity Joins in Data Cleaning [C] // in: Data Engineering, 2006 ICDE ' 06 Proceedings of the 22nd International Conference on, 2006: 5-5.
[47] XIAO C, WANG W, LIN X, et al. Efficient similarity joins for near duplicate detection [C] // in: Proceedings of the 17th international conference on World Wide Web, Beijing, China. 1367516: ACM, 2008: 131-140.
[48] PAPAPETROU P, ATHITSOS V, KOLLIOS G, et al. Reference-based alignment in large sequence databases [J]. Proc VLDB Endow, 2009, 2(1):205-216.
[49] LI C, LU J, LU Y. Efficient Merging and Filtering Algorithms for Approximate String Searches [C] // in: Proceedings of the 2008 IEEE 24th International Conference on Data Engineering. 1547171: IEEE Computer Society, 2008: 257-266.
[50] LI C, WANG B, YANG X. VGRAM: improving performance of approximate queries on string collections using variable-length grams [C] // in: Proceedings of the 33rd international conference on Very large data bases, Vienna, Austria. USA: VLDB Endowment, 2007: 303-314.
[51] YANG X, WANG B, LI C. Cost-based variable-length-gram selection for string collections to support approximate queries efficiently [C] // in: Proceedings of the 2008 ACM SIGMOD international conference on Management of data, Vancouver, Canada. 1376655: ACM, 2008: 353-364.
[52] BEHM A, SHENGYUE J, CHEN L, et al. Space-Constrained Gram-Based Indexing for Efficient Approximate String Search [C] // in: Data Engineering, 2009 ICDE ' 09 IEEE 25th International Conference on Data Engineering, USA. USA: IEEE, 2009: 604-615.
[53] 邱越峰, 田增平, 季文云, 周傲英. 一种高效的检测相似重复记录的方法 [J]. 计算机学报, 2001, 24(1):69-77. (QIU Yue-Feng, TIAN Zeng-Ping, JI Wen-Yun, ZHOU Ao-Ying. An Efficient Approach for Detecting Approximately Duplicate Database Records. Chinese Journal of Computers, 2001, 24(1):69-77.)
[54] LIEBERMAN M D, SANKARANARAYANAN J, SAMET H. A Fast Similarity Join Algorithm Using Graphics Processing Units [C] // in: Proceedings of the 2008 IEEE 24th International Conference on Data Engineering, USA. USA: IEEE Computer Society, 2008: 1111-1120.

[55] 燕彩蓉, 万永权. 并行实体解析与记录聚合模型 [J]. 小型微型计算机系统, 2013, (08):1843-1847. (YAN Cai-rong, Wan Yong-quan. Parallel Entity Resolution and Record Aggregation Model. Journal of Chinese Computer Systems.)
[56] 燕彩蓉, 张洋舜, 徐光伟. 支持隐私保护的众包实体解析 [J]. 计算机科学与探索, 2014, (07):802-811. (YAN Cairong, ZHANG Yangshun, XU Guangwei. Crowdsourcing entity resolution with privacy protection. Journal of Frontiers of Computer Science and Technology, 2014, 8(7):802-811.)
[57] 王宁, 李杰. 大数据环境下用于实体解析的两层相关性聚类方法 [J]. 计算机研究与发展, 2014, (09):2108-2116. (Wang Ning and Li Jie. Two-Tiered Correlation Clustering Method for Entity Resolution in Big Data. Journal of Computer Research and Development, 2014, (09):2108-2116.)
[58] 杨丹, 申德荣, 于戈, et al. 数据空间中时间为中心的集合实体识别策略 [J]. 计算机科学与探索, 2012, 6(11):974-984. (YANG Dan, SHEN Derong, YU Ge, et al. Time-centered collective entity resolution strategy in dataspace. Journal of Frontiers of Computer Science and Technology, 2012, 6(11):974-984.)

Note: Figure translations are in progress. See original paper for figures.