ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

*(A High Impact Factor, Monthly, Peer Reviewed Journal)*

*Website: www.ijircce.com*

**Vol. 7, Issue 12, December 2019**

# Computing the Normalized Record with Record, Field, and Value-Component Granularity Levels

Ms. Sonali Pednekar [1], Prof. Harish Barapatre [2] , Prof. Ankit Sanghvi [3]

Dept. of Computer Engineering, Alamuri Ratnamala Institute of Engineering and Technology, Maharashtra, India [1,3]

Dept. of Computer Engineering, Yadavrao Tasgaonkar Institute of Engineering and Technology, Maharashtra, India [2]

**ABSTRACT:** In the proposed system the problem is to record normalization over a set of matching records that refer to the same real-world entity. We presented three levels of normalization granularities (record-level, field-level and value component level) and two forms of normalization (typical normalization and complete normalization).We propose three levels of granularities for record normalization along with methods to construct normalized records according to them. We propose a comprehensive framework for systematic construction of normalized records. Our framework is flexible and allows new strategies to be added with ease. To our knowledge, this is the first piece of work to propose such a detailed framework. We propose and compare a range of normalization strategies, from frequency, length, centroid and feature-based to more complex ones that utilize result merging models from information retrieval, such as (weighted) Borda. We introduce a number of heuristic rules to mine desirable value components from a field. We use them to construct the normalized value for the field. We perform empirical studies on publication records.

**KEYWORDS:** Record normalization, data quality, data fusion, web data integration.

## I. INTRODUCTION

We have witnessed the rapid growth of the World Wide Web– The Web has not only "broadened" but also "deepened": A 1999 survey [1] estimated a total of 800 million pages on the Web at that time. Nowadays, 19.2 billion Web pages as reported by the recent index of Yahoo.com, denoting at least 24 times increase in six years. Further, while this surface Web has linked billions of static HTML pages, an equally or even more significant amount of information is "hidden" on the deep Web, behind the query forms of searchable databases. A July 2000 survey [2] estimated 96,000 "search sites" and 550 billion content pages in this deep Web. A more recent study [3] in April 2004 estimated 330,000 deep Web sources with over 1.2 million query forms, reflecting a fast 3-7 times increase in 4 years. With the virtually unlimited amount of unstructured and structured information sources, the Web is clearly an important frontier for data management and knowledge discovery.

Accessing information on the Web thus requires not only search to locate pages of interests, on the surface Web, but also integration to aggregate data from alternative or complementary sources, on the deep Web. While the opportunities are unprecedented, the challenges are also immense: For the surface Web, while search seems to have evolved into a standard technology, its maturity and pervasiveness have invited the attack of spam and the demand of personalization. On the other hand, for the deep Web, while the proliferation of structured sources has promised opportunities for more precise and aggregated access, it has presented new challenges for large scale and dynamic information integration. These issues are essentially related to data management, in a large scale, and thus present novel problems and interesting opportunities for our research community.

The Web has evolved into a data-rich repository containing a large amount of structured content spread across millions of sources. The usefulness of Web data increases exponentially (e.g., building knowledge bases, Web-

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

*(A High Impact Factor, Monthly, Peer Reviewed Journal)*

*Website: www.ijircce.com*

**Vol. 7, Issue 12, December 2019**

scale data analytics) when it is linked across numerous sources. Structured data on the Web resides in Web databases [1] and Web tables [2]. Web data integration is an important component of many applications collecting data from Web databases, such as Web data warehousing (e.g., Google and Bing Shopping; Google Scholar), data aggregation (e.g., product and service reviews), and meta searching. We propose three levels of granularities for record normalization along with methods to construct normalized records according to them.

## II. REVIEW OF LITERATURE

Integration systems at Web scale need to automatically match records from different sources that refer to the same real-world entity [4], [5], [6], find the true matching records among them and turn this set of records into a standard record for the consumption of users or other applications. There is a large body of work on the record matching problem [7] and the truth discovery problem [8]. The record matching problem is also referred to as duplicate record detection [9], record linkage [10], object identification [11], entity resolution, or deduplication and the truth discovery problem is also called as truth finding or fact finding a key problem in data fusion. In this report, we assume that the tasks of record matching and truth discovery have been performed and that the groups of true matching records have thus been identified. Our goal is to generate a uniform, standard record for each group of true matching records for end-user consumption. We call the generated record the normalized record. We call the problem of computing the normalized record for a group of matching records the record normalization problem (RNP), and it is the focus of this work. RNP is another specific interesting problem in data fusion.

The promise of Big Data hinges upon addressing several big data integration challenges, such as record linkage at scale, real-time data fusion, and integrating Deep Web. Although much work has been conducted on these problems, there is limited work on creating a uniform, standard record from a group of records corresponding to the same real-world entity. He refer to this task as record normalization. Such a record representation, coined normalized record, is important for both front-end and back-end applications.

**Chang and J. Cho, he has** formulated and solved the query planning and optimization problem for deep web databases with dependencies. We have developed a dynamic query planner with an approximation algorithm with a provable approximation ratio of 1/2. He have also developed cost models to guide the planner. The query planner automatically selects best sub-goals on-the-fly. The K query plans generated by the planner can provide alternative plans when the optimal one is not feasible. Our experiments show that the cost model for query planning is effective. Despite using an approximate algorithm, our planning algorithm outperforms the naive planning algorithm, and obtains the optimal query plans for most experimental queries in terms of both number of databases involved and actual execution time. He also shows that our system has good scalability.

## III. OBJECTIVE

Relevant objective of the proposed system are as follows

1. To record normalization along with methods to construct normalized records according to them.
2. To propose a comprehensive framework for systematic construction of normalized records.
3. To compare a range of normalization strategies, from frequency, length, centroid and feature-based to more complex ones that utilize result merging models from information retrieval, such as (weighted) Borda.
4. To introduce a number of heuristic rules to mine desirable value components from a field
5. To study the system on publication records.

In this system our aim to develop a framework for constructing normalized records systematically. We propose three levels of normalization: record, field, and value component. We introduce a number of heuristic rules to mine desirable value components from a field. We use them to construct the normalized value for the field.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

*(A High Impact Factor, Monthly, Peer Reviewed Journal)*

*Website: www.ijircce.com*

**Vol. 7, Issue 12, December 2019**

## IV. PROBLEM DEFINITION

Record normalization is important in many application domains. For example, in the research publication domain, although the integrator website, such as Citeseer or Google Scholar, contains records gathered from a variety of sources using automated extraction techniques, it must display a normalized record to users. Otherwise, it is unclear what can be presented to users:

1. Present the entire group of matching records or
2. Simply present some random record from the group, to just name a couple of adhoc approaches.

Either of these choices can lead to a frustrating experience for a user, because in (1) the user needs to sort/browse through a potentially large number of duplicate records, and in (2) we run the risk of presenting a record with missing or incorrect pieces of data.

Record normalization is a challenging problem because different Web sources may represent the attribute values of an entity in different ways or even provide conflicting data. Conflicting data may occur because of incomplete data, different data representations, missing attribute values, and even erroneous data.

| Fields | author | title | venue | date | pages |
|---|---|---|---|---|---|
| $R_a$ | Halevy, A.; Rajaraman A.; Ordille, J. | Data integration: the teenage years | in proc 32nd int conf on Very large data bases | 2006 | |
| $R_b$ | A. Halevy, A. Rajaraman, J. Ordille | Data integration: the teenage years | in VLDB | 2006 | 9-16 |
| $R_c$ | A. Halevy, A. Rajaraman, J. Ordille | Data integration: the teenage years | in proc 32nd conf on Very large data bases | 2006 | pp.9-16 |
| $R_d$ | A. Halevy, A. Rajaraman, J. Ordille | Data integration: the teenage years | | 2006 | 9-16 |
| $R_{norm}$ | Alon Halevy, Anand Rajaraman, Joann Ordille | Data integration: the teenage years | in proceedings of the 32nd international conference on Very large data bases | 2006 | 9-16 |
| $R_{field}$ | A. Halevy, A. Rajaraman, J. Ordille | Data integration: the teenage years | in proc 32nd int conf on Very large data bases | 2006 | pp.9-16 |

**TABLE 1: Four records for the same publication: $R_a$, $R_b$, $R_c$, and $R_d$ are extracted from different websites and Rnorm is constructed manually.**

For example, Table 1 contains four records corresponding to the same entity (publication). They are extracted from different websites. Record $R_{norm}$ is constructed by hand for illustration purposes. One notices that the same publication has different representations in different websites. For instance, the field author uses the format ”last-name, first-name-initial” in the record $R_a$, but the values of the same field in the records $R_b$, $R_c$, and $R_d$ use the format ”first-name-initial. last-name”. One can also observe that the value of the field pages is absent in Ra. The field venue has incomplete values in three of the four records and has no value in $R_d$; it contains the abbreviations “proc”, “int”, “conf” to represent “proceedings”, “international” and “conference”, respectively, in the records $R_a$ and $R_c$; it contains the acronym “VLDB” to represent “Very Large Data Bases” while missing “proceedings of the 32nd international conference on” in $R_b$. Some values of the attributes of Rnorm cannot be acquired directly from the given set of matching records, such as the first names of the authors. They could be obtained by mining external sources, such as a search engine. In this report, we focus on the best effort record normalization: we compute $R_{norm}$ from the set of matching records and do not explore external sources. Furthermore, this report only focuses on the normalization of text data.

## V. PRAPOSED SYSTEM

We identify three levels of normalization granularity: record, field, and value-component.

**Record level** assumes that the values of the fields within a record are governed by some hidden criterion and that together create a cohesive unit that is user-friendly. As a consequence, this normalization favors building the normalized record from entire records among the set of matching records rather than piecing it together from field values of different records. Thus, any of the matching records (ideally, that has no missing values) can be the normalized record.

**Field level** assumes that record level is often inadequate in practice because records contain fields with incomplete values. Recall that these records are the products of automatic data extraction tools, which are not perfect and thus may

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 7, Issue 12, December 2019

produce errors. This normalization level ignores the cohesion factor in the record normalization level and assumes that a user is better served when each field of the normalized record has as easy to understand a value as possible, selected from among the values in the set of matching records. It treats each field of the normalized record independently, finds a normalized value (according to some criterion) per field, and creates the normalized record by stitching together the normalized values of the fields. The normalized record may not resemble any of the matching records, but it will convey the same information as any of them, in a user-friendlier form than any of the individual records.

**Value-component level** takes the field level normalization a step"deeper." It assumes that in general the value of a field may comprise of multiple pieces some of which may not be easy to grasp by an ordinary user. For example, a field (such as venue) may contain arcane acronyms illegible to an ordinary user. A normalization solution in accordance with this level will yield a value for a field with the property that the individual components of the value are themselves normalized. The resulted (normalized) value may not physically exist in any of the matching records.
Thus, we can create a normalized value for venue, at the value-component level, as follows.
1) We take the value suggested previously by the field level for venue and replace the abbreviations in it with the complete words and change it into "in proceedings 32nd international conference on Very large data bases".
2) We find that "in proceedings" is the part of the collocation "in proceedings of the".
3) We use the collocation to replace "in proceedings".
4) Finally, we get the normalized value of venue, "in proceedings of the 32nd international conference on Very large data bases".

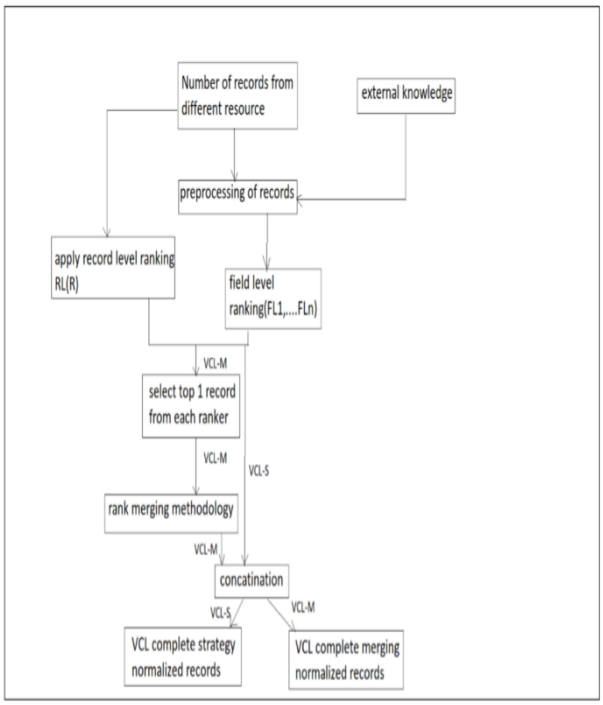


*Figure N0-1 system Architecture*

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 7, Issue 12, December 2019

METHODS:

**1. Ranking-based Strategies**

We utilize four ranking strategies: frequency, length, centroid, and feature-based. We use them to construct several rankers. at record and field levels. To give a uniform presentation, we refer to records and their fields as units in this section.

**2. Value Component Mining**

Many common value components of a field are abbreviations, which need to be expanded to improve the readability of the normalized record. For example, in the field venue, "proc" is often used to represent "proceedings". The sub-collocation relation is a useful tool to organize the components of the values of a field in a partial order and then identify a template collocation from them. For example, "in proceedings of the" is a template collocation, but it often times takes the form of sub collocations such as "proceedings of", "proceedings of the" and "in proceedings", which should be replaced with the template collocation. Template collocations tend to co-occur frequently. For example, "conference on" frequently co-occurs with "in proceedings of the".

**3. Ranked List Merging**

We introduced a set of single-strategy rankers each of which ranks the units (records or field values) with a different strategy. In general, a single-strategy approach does not produce satisfactory results and may even cause bias. We utilize a multi-strategy approach to combine the outcomes of several single-strategy rankers to overcome the limitations of the individual rankers. A multi-strategy approach requires an effective rank merging algorithm.

## VI. SECURITY ANALYSIS

### VI. Algorithm

**Mining Abbreviations-Definition pairs**

**Input:** $Val(f_j) = \{ r_i[f_j] \mid r_i \in R^e \}$ : the collection of all values of the field $f_j$

**Output:** AWP: a set of abbreviation-word pairs

1. cwords = ∅; AWP = ∅;
2. pwords = tokenize($Val(f_j)$)
3. uwords = unique(pwords);
4. for each uword ∈ uwords do
5. if len(uword) ≥ $\eta_{len}$ and idf (uword,$R^e$) ≤ $\eta_{idf}$ then
6. insert uword into cwords;
7. end if
8. end for
9. for each cword ∈ cwords do
10. pa_words = getWordsBySameContext( cword, uwords, $\eta_{pos}$);
11. if pa_words != ∅ then
12. abbreviations = getAbbreviations( cword, pa_words);
13. end if
14. if abbreviations != ∅ then
15. for each abbreviation ∈ abbreviations do
16. insert (abbreviation, cword) into AWP;
17. end for
18. end if
19. end for
20. return AWP

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

*(A High Impact Factor, Monthly, Peer Reviewed Journal)*

*Website: www.ijircce.com*

**Vol. 7, Issue 12, December 2019**

**SOFTWARE REQUIRMENT**

1. Operating System: Windows 7 and above.
2. IDE: Netbeans 8.2
3. Programming Language : Java Programming
4. Database: MySQL 5.5
5. Toolkit: JDK 1.8

## VII. IMPLEMENTATION

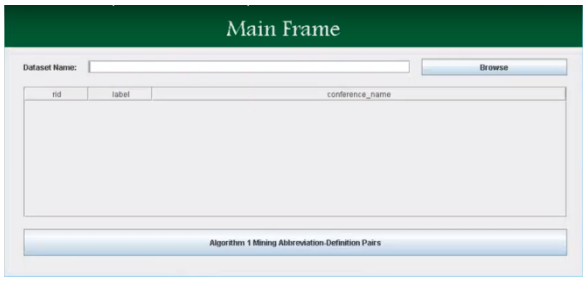


**Figure No-2: Main Frame with importing dataset**

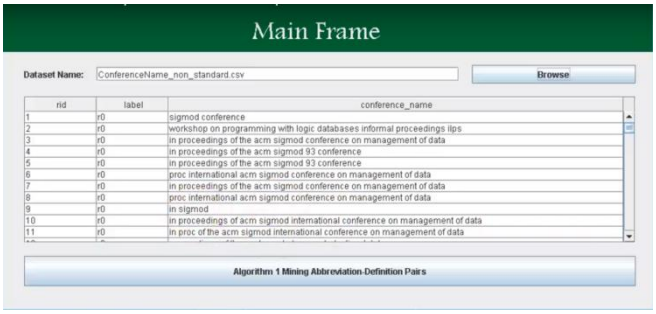


**Figure No-3: Main Frame with preprocess dataset**

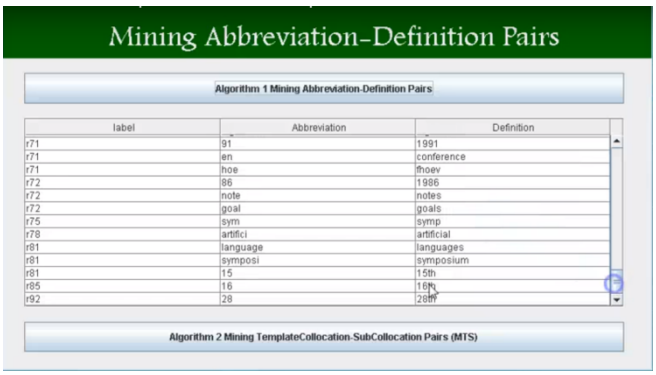


**Figure No-4: Output of Algorithm mining definition Pair.**

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

*(A High Impact Factor, Monthly, Peer Reviewed Journal)*

*Website: www.ijircce.com*

**Vol. 7, Issue 12, December 2019**

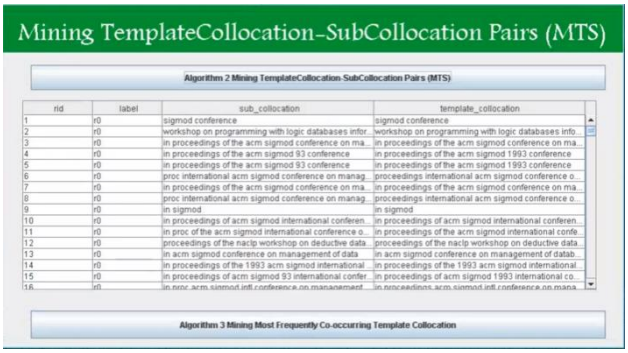


**Figure No-5: Output of MTS Algorithm.**

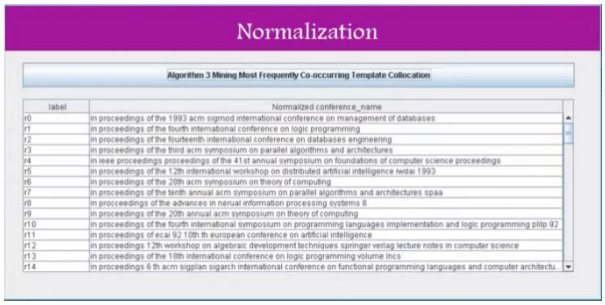


**Figure No-6: Final Normalization Form.**

## VIII. RESULTS AND DISCUSSION

We use the dataset PVCD. The dataset contains data about publication venue canonicalization. PVCD has 3,683 publication venue values for 100 distinct real-world publication records. It is only concerned with the field venue, which is arguably the most difficult field to normalize, because of the presence of acronyms, abbreviations, and misspellings. We use this dataset to compare our approaches with those in. The work in is an instance of typical normalization, because it selects one of the duplicate records or one of the field values as the normalized record or field value, respectively. It does not attempt to create new field values or new records as normalized records. Our analysis of the dataset reveals that many normalized field values are labeled unreasonably.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

*(A High Impact Factor, Monthly, Peer Reviewed Journal)*

*Website: www.ijircce.com*

**Vol. 7, Issue 12, December 2019**

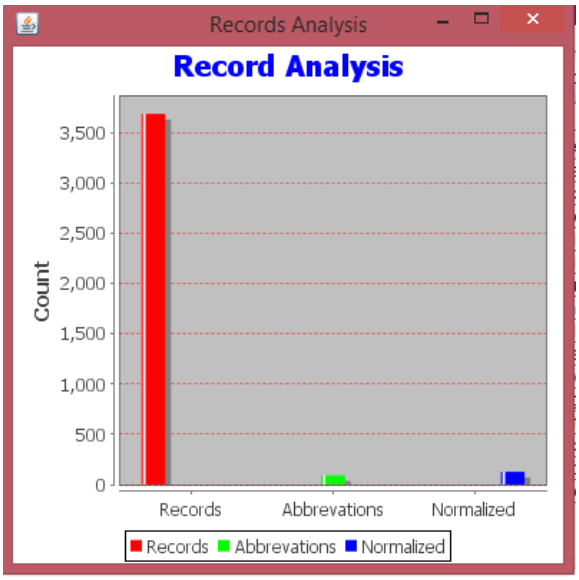


**Figure No-7: Record Analysis.**

We have processed total 3568 record of PVCD data set also our proposed system processed this data and gives abbreviated record is 92 and finally normalized records shows 127 record that means our system removes duplicate record and shows approximate 0.83 accuracy of normalized records.

## IX. CONCLUSION

In this system, we formalize the record normalization problem, present in-depth analysis of normalization granularity levels (e.g., record, field, and value-component) and of normalization forms (e.g., typical versus complete). We introduce a number of heuristic rules to mine desirable value components from a field. We use them to construct the normalized value for the field. We perform empirical studies on publication records. We propose and compare a range of normalization strategies, from frequency, length, centroid and feature-based to more complex ones that utilize result merging models from information retrieval, such as (weighted) Borda. In the future, we plan to extend our research as follows. First, conduct additional experiments using more diverse and larger datasets. The lack of appropriate datasets currently has made this difficult. Second, investigate how to add an effective human-in-the-loop component into the current solution as automated solutions alone will not be able to achieve perfect accuracy. Third, develop solutions that handle numeric or more complex values.

## REFERENCES

[1] K. C.-C. Chang and J. Cho, Accessing the web: From search to integration, in SIGMOD, 2006, pp. 804805.
[2] M. J. Cafarella, A. Halevy, D. Z. Wang, E. Wu, and Y. Zhang, Web tables: Exploring the power of tables on the web, PVLDB, vol. 1, no. 1, pp. 538549, 2008.
[3] W. Meng and C. Yu, Advanced Meta search Engine Technology. Morgan & Claypool Publishers, 2010.
[4] A. Gruenheid, X. L. Dong, and D. Srivastava, Incremental record linkage, PVLDB, vol. 7, no. 9, pp. 697708, May 2014.
[5] E. K. Rezig, E. C. Dragut, M. Ouzzani, and A. K. Elmagarmid, Query-time record linkage and fusion over web databases, in ICDE, 2015, pp. 4253.
[6] W. Su, J. Wang, and F. Lochovsky, Record matching over query results from multiple web databases, TKDE, vol. 22, no. 4, 2010.
[7] H. Kopcke and E. Rahm, Frameworks for entity matching: A comparison, DKE, vol. 69, no. 2, pp. 197210, 2010.
[8] X. Yin, J. Han, and S. Y. Philip, Truth discovery with multiple conflicting information providers on the web, ICDE, 2008.
[9] A. K. Elmagarmid, P. G. Ipeirotis, and V. S. Verykios, Duplicate record detection: A survey, TKDE, vol. 19, no. 1, pp. 116, 2007

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

*(A High Impact Factor, Monthly, Peer Reviewed Journal)*

*Website: www.ijircce.com*

**Vol. 7, Issue 12, December 2019**

[10] X. L. Dong and F. Naumann, Data fusion: resolving data conicts for integration, PVLDB, vol. 2, no. 2, pp. 16541655, 2009.
[11] E. K. Rezig, E. C. Dragut, M. Ouzzani, A. K. Elmagarmid, and W.G.Aref
[12] X. Wang, X. L. Dong, and A. Meliou, Data x-ray: A diagnostic tool for data errors, in SIGMOD, 2015, pp. 12311245.
[13] G. R. D. Patrick AV Hall, Approximate string matching, ACM Computing Surveys, vol. 12, no. 4, pp. 381402, 1980.
[14] W. W. Cohen, P. Ravikumar, and S. E. Fienberg, A comparison of string metrics for matching names and records, in KDD workshop on data cleaning and object consolidation, 2003, pp. 7378.
[15] D. C. Liu and J. Nocedal, On the limited memory bfg method for large scale optimization, Mathematical Programming, vol. 45, no. 3, pp. 503528, 1989.