Data Cleaning

A Practical Perspective

Synthesis Lectures on Data Management

Foundations of Data Quality Management
Wenfei Fan and Floris Geerts
2012

Incomplete Data and Data Dependencies in Relational Databases
Sergio Greco, Cristian Molinaro, and Francesca Spezzano
2012

Business Processes: A Database Perspective
Daniel Deutch and Tova Milo
2012

Data Protection from Insider Threats
Elisa Bertino
2012

Deep Web Query Interface Understanding and Integration
Eduard C. Dragut, Weiyi Meng, and Clement T. Yu
2012

P2P Techniques for Decentralized Applications
Esther Pacitti, Reza Akbarinia, and Manal El-Dick
2012

Query Answer Authentication
HweeHwa Pang and Kian-Lee Tan
2012

Declarative Networking
Boon Thau Loo and Wenchao Zhou
2012

Full-Text (Substring) Indexes in External Memory
Marina Barsky, Ulrike Stege, and Alex Thomo
2011

Spatial Data Management
Nikos Mamoulis
2011

Database Repairing and Consistent Query Answering
Leopoldo Bertossi
2011

Managing Event Information: Modeling, Retrieval, and Applications
Amarnath Gupta and Ramesh Jain
2011

Fundamentals of Physical Design and Query Compilation
David Toman and Grant Weddell
2011

Methods for Mining and Summarizing Text Conversations
Giuseppe Carenini, Gabriel Murray, and Raymond Ng
2011

Probabilistic Databases
Dan Suciu, Dan Olteanu, Christopher Ré, and Christoph Koch
2011

Peer-to-Peer Data Management
Karl Aberer
2011

Probabilistic Ranking Techniques in Relational Databases
Ihab F. Ilyas and Mohamed A. Soliman
2011

Uncertain Schema Matching
Avigdor Gal
2011

Fundamentals of Object Databases: Object-Oriented and Object-Relational Design
Suzanne W. Dietrich and Susan D. Urban
2010

Advanced Metasearch Engine Technology
Weiyi Meng and Clement T. Yu
2010

Web Page Recommendation Models: Theory and Algorithms
Sule Gündüz-Ögüdücü
2010

Multidimensional Databases and Data Warehousing
Christian S. Jensen, Torben Bach Pedersen, and Christian Thomsen
2010

Database Replication
Bettina Kemme, Ricardo Jimenez-Peris, and Marta Patino-Martinez
2010

Relational and XML Data Exchange
Marcelo Arenas, Pablo Barcelo, Leonid Libkin, and Filip Murlak
2010

User-Centered Data Management
Tiziana Catarci, Alan Dix, Stephen Kimani, and Giuseppe Santucci
2010

Data Stream Management
Lukasz Golab and M. Tamer Özsu
2010

Access Control in Data Management Systems
Elena Ferrari
2010

An Introduction to Duplicate Detection
Felix Naumann and Melanie Herschel
2010

Privacy-Preserving Data Publishing: An Overview
Raymond Chi-Wing Wong and Ada Wai-Chee Fu
2010

Keyword Search in Databases
Jeffrey Xu Yu, Lu Qin, and Lijun Chang
2009

Data Cleaning: A Practical Perspective

Venkatesh Ganti and Anish Das Sarma

www.morganclaypool.com

ISBN: 9781608456772 paperback
ISBN: 9781608456789 ebook

DOI 10.2200/S00523ED1V01Y201307DTM036

A Publication in the Morgan & Claypool Publishers series
SYNTHESIS LECTURES ON DATA MANAGEMENT

Lecture #36
Series Editor: M. Tamer Özsu, *University of Waterloo*
Series ISSN
Synthesis Lectures on Data Management
Print 2153-5418 Electronic 2153-5426

Data Cleaning

A Practical Perspective

Venkatesh Ganti
Alation Inc.

Anish Das Sarma
Google Inc.

SYNTHESIS LECTURES ON DATA MANAGEMENT #36

MORGAN & CLAYPOOL PUBLISHERS

ABSTRACT

Data warehouses consolidate various activities of a business and often form the backbone for generating reports that support important business decisions. Errors in data tend to creep in for a variety of reasons. Some of these reasons include errors during input data collection and errors while merging data collected independently across different databases. These errors in data warehouses often result in erroneous upstream reports, and could impact business decisions negatively. Therefore, one of the critical challenges while maintaining large data warehouses is that of ensuring the quality of data in the data warehouse remains high. The process of maintaining high data quality is commonly referred to as *data cleaning*.

In this book, we first discuss the goals of data cleaning. Often, the goals of data cleaning are not well defined and could mean different solutions in different scenarios. Toward clarifying these goals, we abstract out a common set of data cleaning tasks that often need to be addressed. This abstraction allows us to develop solutions for these common data cleaning tasks. We then discuss a few popular approaches for developing such solutions. In particular, we focus on an operator-centric approach for developing a data cleaning platform. The operator-centric approach involves the development of customizable operators that could be used as building blocks for developing common solutions. This is similar to the approach of relational algebra for query processing. The basic set of operators can be put together to build complex queries. Finally, we discuss the development of custom scripts which leverage the basic data cleaning operators along with relational operators to implement effective solutions for data cleaning tasks.

KEYWORDS

data cleaning, deduplication, record matching, data cleaning scripts, schema matching, ETL, clustering, record matching, deduplication, data standardization, ETL data flows, set similarity join, segmentation, parsing, string similarity functions, edit distance, edit similarity, jaccard similarity, cosine similarity, soundex, constrained deduplication, blocking

Contents

Preface

Data cleaning is the process of starting with raw data from one or more sources and maintaining reliable quality for your applications. We were motivated to write this book since we found a gap in technical material that clearly explained the goals and capabilities of a data cleaning solution; in general, data cleaning is usually thought of as a solution for an individual problem. One of the prominent issues we had was that there was no guide offering practical advice on options available for building or choosing a data cleaning solution. In this book, we fill this gap.

Our approach toward this book was to conceptualize data cleaning solutions as being composed of *tasks* and *operators*. Each solution is a composition of multiple high-level tasks, and each task may have one or more operator-based solutions. In this book we elaborate on the most common tasks, and their implementations leveraging critical operators. Our book can be seen as a practitioner's guide to understand the space of options for evaluating or building a good data cleaning solution. We provide an overview of the capabilities required in such a system, which are the set of tasks described in this book. People building complete solutions may use the set of tasks described here, and choose from the space of operators. Therefore, this book is ideally suited for practitioners of data cleaning and students interested in the topic. Although our book lists the useful tools, techniques, and pointers, some of them require custom implementations with no open-source components available. Therefore, if students or engineers are looking for good abstractions for plugins to build, we hope that our book provides some options.

For beginners interested in data cleaning, we suggest reading the material sequentially from the first chapter. Advanced readers may directly jump to any relevant chapter for reference; each chapter is self contained and provides further pointers to existing research.

We enjoyed writing this book and gained new insights in the process that we've shared in this material. We sincerely wish we had more time, in which case we would have been able to add more depth on several directly related topics. For example, the user-interface aspects of data cleaning have not received due attention in this book.

Venkatesh Ganti and Anish Das Sarma
September 2013

Acknowledgments

The structure and material in this book has been significantly influenced by people that both of us have worked with closely on several topics related to data cleaning. Prominently, some of them are Surajit Chaudhuri, Raghav Kaushik, Arvind Arasu, Eugene Agichtein, and Sunita Sarawagi. We are grateful to Tamer Ozsu and the publisher for the opportunity to explain our view on data cleaning in this book.

We also thank our families for their patience while we spent long hours outside of work writing this book, which should have been devoted to them instead.

Venkatesh Ganti and Anish Das Sarma
April 2013

CHAPTER 1

Introduction

Databases are ubiquitous in enterprise systems, and form the backbone for systems keeping track of business transactions and operational data. They also have become the defacto standard for supporting data analysis tasks generating reports indicating the health of the business operations. These reports are often critical to track performance as well as to make informed decisions on several issues confronting a business. The reporting functionality has become so important on its own that businesses often create consolidated data repositories. These repositories can be observed in several scenarios such as data warehousing for analysis, as well as for supporting sophisticated applications such as comparison shopping.

1.1 ENTERPRISE DATA WAREHOUSE

Data warehouses are large data repositories recording interactions between various entities that an enterprise deals with: customers, products, geographies, etc. By consolidating most of the relevant data describing the interactions into one repository, data warehouses facilitate canned and adhoc data analysis over such interactions.

The results of such analysis queries often form the backbone of several critical reports, which help evaluate and monitor performance of various business projects. These reports may often be useful for prioritizing among various business initiatives. Therefore, accuracy of data in these data warehouses is critical. Errors in these databases can result in significant downstream reporting errors. Sometimes, such errors can result in bad decisions being taken by the executives.

Errors in data tend to creep in from a variety of sources, say when new sales records are inserted. For instance, enterprises routinely obtain resellers' sales interactions with customers from resellers. Data entry at the point of sales is often performed in a rush and causes many errors in data. Sometimes, these errors are introduced because the sales agent does not try to find out the correct data, and enters a default or a typical value. So, the data about the customer sent by the reseller may not match with the current record in the data warehouse.

Alternatively, a large number of errors are often introduced into the data warehouse when data from a new source database is merged with it. Such data consolidation is required when sales transactions from a new data feed (say, an OLTP database) are inserted into the data warehouse. If some of the new records in both the source and target describe the same entities, then it is often possible that the data merger results in several data quality issues because interactions with the same entity are now distributed across multiple records.

1.2 COMPARISON SHOPPING DATABASE

Many popular comparison shopping search engines (e.g., Bing Shopping, Google Products, ShopZilla) are backed by comprehensive product catalog and offer databases consisting, respectively, of products and offers from multiple merchants to sell them at specific prices. The catalog and offer databases enable a comparison shopping engine to display products relevant to a user's search query and for each product the offers from various merchants. These databases are populated and maintained by assimilating feeds from both catalog providers (such as CNet, PriceGrabber) as well as from merchants (e.g., NewEgg.com, TigerDirect.com). These feeds are consolidated into a master catalog along with any other information per product received from merchants or from other sources. When a user searches for a product or a category of products, these comparison shopping sites display a set of top-ranking items for the specific user query. When a user is interested in a specific product, the user is then shown the list of merchants along with offers for each of them.

These product catalog and merchant feeds are obtained from independently developed databases. Therefore, identifiers and descriptions of the same product and those in the corresponding offers will very likely be different across each of the input feeds. Reconciling these differences is crucial for enabling a compelling useful comparison shopping experience to a user. Otherwise, information about the same product would be split across multiple records in the master catalog. Whichever record is shown to the user, the user is only shown a part of the information in the master catalog about the product. Therefore, one of the main goals is to maintain a correctly consolidated master catalog where each product sold at several merchants has only one representation.

Similar data quality issues arise in the context of *Master Data Management (MDM)*. The goal of an MDM system is to maintain a unified view of non-transactional data entities (e.g., customers, products) of an enterprise. Like in the data warehousing and comparison shopping scenarios, these master databases often grow through incremental or batch insertion of new entities. Thus, the same issues and challenges of maintaining a high data quality also arise in the context of master data management.

1.3 DATA CLEANING TASKS

Data cleaning is an overloaded term, and is often used loosely to refer to a variety of tasks aimed at improving the quality of data. Often, these tasks may have to be accomplished by stitching together multiple operations. We now discuss some common data cleaning tasks to better understand the underlying operations. We note that this list includes commonly encountered tasks, and is not comprehensive.

1.4 RECORD MATCHING

Informally, the goal of record matching is to match each record from a set of records with records in another table. Often, this task needs to be accomplished when a new set of entities is imported to the target relation to make sure that the insertion does not introduce duplicate entities in the target relation.

Enterprise Data Warehousing Scenario: Consider a scenario when a new batch of customer records is being imported into a sales database. In this scenario, it is important to verify whether or not the same customer is represented in both the existing as well as the incoming sets and only retain one record in the final result. Due to representational differences and errors, records in both batches could be different and may not match exactly on their key attributes (e.g., name and address or the CustomerId). The goal of a record matching task is to identify record pairs, one in each of two input relations, which correspond to the same real-world entity. Challenges to be addressed in this task include (i) identification of criteria under which two records represent the same real-world entity, and (ii) efficient computation strategies to determine such pairs over large input relations.

Table 1.1: Two sets of customer records

ID	Name	Street	City	Phone
r1	Sweetlegal Investments Inc	202 North	Redmond	425-444-5555
r2	ABC Groceries Corp	Amphitheatre Pkwy	Mountain View	4081112222
r3	Cable television services	One Oxford Dr	Cambridge	617-123-4567
s1	Sweet legal Invesments Incorporated	202 N	Redmond	
s2	ABC Groceries Corpn.	Amphitheetre Parkway	Mountain View	
s3	Cable Services	One Oxford Dr	Cambridge	6171234567

Comparison Shopping Scenario: Recall the comparison shopping scenario, where the target comparison shopping site maintains a master catalog of products. Suppose a merchant sends a new feed of products, as shown in Table 1.2. Each of these products has to be matched with a target in the master, or if there is no such matching product, add it as a new product to the master catalog.

Ideally, the merchant could also send a unique identifier that matches a global identifier in the master catalog. In the case of books, ISBN is an identifier that everyone agrees to and uses. However, in other categories of products, there is no such global identifier that can be used for matching. The main challenge here is that the description often used by the merchant may not match with the description at the target comparison shopping site. Hence, matching products "correctly" requires several challenges to be addressed.

The hardness is further exacerbated in the case of products where the underlying product description is often a concatenation of several attribute values. The individual values may them-

selves be equal while the concatenation of these values in different orders could cause the two strings to look very different.

Table 1.2: Product catalog with a new set of products

ID	Title
r1	Canon EOS 20D Digital SLR Body Kit (Req. Lens) USA
r2	Nikon D90 SLR
s1	Canon EOS 20d Digital Camera Body USA - Lens sold separately
s2	Nikon D90 SLR Camera

Record matching is discussed further in Chapter 7.

1.5 SCHEMA MATCHING

A task that often precedes record matching is that of *Schema Matching*: the task of aligning attributes from different schemas. As an example, suppose the information from our warehouse example were organized as a relation $R(Name, CityAddress, Country, Phone, \ldots)$, which stores most of the address (except Country) in a single attribute in textual format. Now suppose you obtain another relation with data represented in the format $S(Company, Apt, Street, City, Zip, Nation, PhoneNumber)$, which breaks the address into individual components. To populate tuples in S into R, we need a process to *convert* each S tuple into the format of R. Schema matching provides: (1) *attribute correspondences* describing which attributes in S correspond to attributes in R; e.g., Country corresponds to Nation, PhoneNumber corresponds to Phone, Company corresponds to Name, and City Address corresponds to the remaining four attributes in S (2) *transformation functions* give concrete functions to obtain attribute values in R from attribute values in S; e.g., a transformation process gives a mechanism to concatenate all attributes to form City Address (or extract attributes like Zip code when converting R to S).

Schema matching is discussed along with record matching in Chapter 7.

1.6 DEDUPLICATION

The goal of *deduplication* is to group records in a table such that each group of records represents the same entity. The deduplication operation is often required when a database is being populated or cleaned the first time.

Informally, the difference between deduplication and record matching is that deduplication involves an additional grouping of "matching" records, such that the groups collectively partition the input relation. Since record matching is typically not transitive (i.e., record pairs $(r1, r2)$

and $(r2, r3)$ may be considered matches but $(r1, r3)$ may not be), the grouping poses additional technical challenges.

For example, consider the enterprise data warehousing scenario. When the data warehouse is first populated from various feeds, it is possible that the same customer could be represented by multiple records in one feed, and even more records across feeds. So, it is important for all records representing the same customer to be reconciled. In Table 1.3, records {g11, g12, g13} are "duplicate" records of each other while {g21, g31} is another set of duplicate records.

Table 1.3: Table showing records with {g11, g12, g13} being one group of duplications, and {g21, g31} another set of duplicate records

ID	Name	Street	City	Phone
g11	Sweetlegal Investments Inc	202 North	Redmond	425-444-5555
g12	Sweet legal Invesments Incorporated	202 N	Redmond	
g13	Sweetlegal Inc	202 N	Redmond	
g21	ABC Groceries Corp	Amphitheatre Pkwy	Mountain View	4081112222
g31	Cable television services	One Oxford Dr	Cambridge	617-123-4567

Let us consider the task of maintaining a shopping catalog in the comparison shopping scenario. Once again, it is possible that a set of records received from a merchant may have multiple records representing the same entity. In the following Table 1.4, {g21, g22, g23, g24} all represent the same entity, a Nikon DSLR camera.

Table 1.4: Table showing grouping, with {g21, g22, g23, g24} all representing the same entity, a Nikon DSLR camera

ID	Title
g1	Canon EOS 20D Digital SLR Body Kit (Req. Lens) USA
g21	Nikon D90 SLR
g22	Nikon D90 SLR Camera
g23	Nikon D90
g24	D90 SLR

Deduplication is discussed in detail in Chapter 8.

1.7 DATA STANDARDIZATION

Consider a scenario where a relation contains several customer records with missing zip code or state values, or improperly formatted street address strings. It is important to fill in missing values and adjust the format of the address strings so as to return correct results for analysis queries. For

instance, if a business analyst wants to understand the number of customers for a specific product by zip code, it is important for all customer records to have the correct zip code values.

The same task is also often required in the maintenance of product catalog databases. For example, ensuring that all dimensions for a set of products are expressed in the same units, and that these attribute values are not missing is very important. Otherwise, search queries on these attributes may not return correct results.

The task of ensuring that all attribute values are "standardized" as per the same conventions is often called *data standardization*.

Data standardization is often a critical operation required before other data cleaning tasks such as record matching or deduplication. Standardizing the format and correcting attribute values leads to significantly better accuracy in other data cleaning tasks such as record matching and deduplication.

1.8 DATA PROFILING

The process of cleansing data is often an iterative and continuous process. It is important to *evaluate* quality of data in a database before one initiates data cleansing process, and subsequently assesses its success. The process of evaluating data quality is called *data profiling*, and typically involves gathering several aggregate data statistics which constitute the data profile. An informal goal of data quality is to ensure that the values match up with expectations. For example, one may expect the customer name and address columns uniquely determine each customer record in a Customer relation. In such a case, the number of unique [name, address] values must be close to that of the total number of records in the Customer relation.

A subset of elements of a data profile may each be obtained using one or more SQL queries. However, the data profile of a database may also consist of a large number of such elements. Hence, computing them all together efficiently is an important challenge here. Also, some of the data profile elements (say, identifying histograms of attribute values which satisfy certain regular expressions) may not easily be computed using SQL queries.

1.9 FOCUS OF THIS BOOK

In this book, we focus our discussion on solutions for data cleaning tasks. However, data cleaning is just one of the goals in an enterprise data management system. For instance, a typical *extract-transform-load (ETL)* process also encompasses several other tasks some of which transform data from sources into the desired schema at the target before merging all data into the target. In this survey, we do not discuss all the goals of ETL.

In particular, we do not discuss the topic of data transformation which is one of the goals of ETL. We also do not discuss the issue of data or information integration, which also requires transforming (perhaps, dynamically) the source data into the schema required by the user's query, besides data cleaning.