Research Statement

Xu Chu

My research is in the general area of data management. Specifically, I have been focusing on tackling the theoretical and the system building challenges associated with managing dirty and inconsistent data. Data errors lead to inaccurate data analytics and wrong business decisions, and have been recognized as the main hurdle in effective data science by the New York Times, Forbes, Gartner, and other major business analysts. For example, poor data across businesses and the government cost the U.S. economy 3.1 trillion per year, according to a report by InsightSquared in 2012 [11]. The advent of big data era further exacerbates the problem of dirty data. Thus, data cleaning has witnessed a surge of interest in both academia and industry. This interest is manifested not only in the large number of research papers in the field, but also in the number of data curation and cleaning startups in the last few years.

There are two major steps in data cleaning: (1) detecting data errors; and (2) repairing the detected errors. The error detection step is often done either through the enforcement of pre-specified data quality rules, or through running statistical analysis, for example, to find outlying values and anomalies. On the other hand, the repairing step often involves running semi-automatic data repairing algorithms to suggest data updates. While error detection is mostly performed using automatic tools, data repairing in practice often requires user involvement, either to verify machine generated updates or to manually fix the data. Both steps incur many technical challenges. In error detection, data quality rules necessary for detecting errors are often not provided, and hence these rules need to be mined from the underlying (possibly dirty) data sets. In addition, most error detection techniques have inherent combinatorial complexity, since detecting errors often requires pair-wise tuple comparisons and other expensive operations. This makes the deployment of most algorithms in big data settings prohibitively expensive. In error repairing, since ground truth is usually not available, principled ways to automatically suggest most likely repairs or to involve humans are necessary. I have made multiple contributions in both areas of data cleaning (detection and repairing) to tackle these challenges, which I will discuss briefly in the following.

In the area of error detection, I have made three main contributions: (1) I have proposed using denial constraints (a fragment of first-order logic) as the formal language to capture a wide range of data quality rules, and I have implemented the RuleMiner system [6, 8] to automatically discover denial constraints from sample data instances. Automatically discovering constraints is particularly useful since asking users to design data quality rules is an expensive process, which requires domain expertise, and is rarely done in practice; (2) I have proposed a novel technique for discovering outliers in certain subsets or contexts of the data [1] which overcomes the shortcomings of previous outlier detection techniques. The discovered contextual outliers are valuable in data exploration and in targeted explanation and diagnosis scenarios; and finally (3) I have designed a distributed algorithm to scale up data deduplication tasks in shared-nothing data processing platforms [4] such as Hadoop and Spark. The proposed distribution strategy has provable theoretical guarantees in terms of both the communication cost and the computation cost.

In the area of data repairing, I have made two main contributions: (1) I have proposed a state-of-the-art holistic error repairing technique [7], which accumulates evidence from a broad spectrum of data quality rules, and suggests more accurate data updates in a holistic manner; and (2) I have explored the use of trustworthy external sources, including curated knowledge bases and human experts, to suggest and verify data updates in a principled way [9, 10].

I have also collaborated with various groups in complementary works in data integration and data management: (1) we have compared various state-of-art error detection techniques on several real world data sets [11]; (2) we have explored automatic extraction of structured data from unstructured data on the Web [2]; (3) we have proposed an automatic technique for performing joins in ad-hoc data analytics [13]; and (4) we

---

1 The research paper is under review
are currently exploring the use of machine learning and inference as a principled way to suggest data repairs. Besides the published research papers that summarize the aforementioned contributions, I have coauthored a survey monograph on data cleaning published in the Foundations and Trends in Databases [14], and I have co-presented tutorials in ACM SIGMOD 2016 [5] and in VLDB 2016 [3] on data cleaning. In addition, I am currently coauthoring a book on data cleaning with my supervisor; the book will be published by ACM Books.

In the following I will give more details on a few of these projects, and I will highlight some future research directions, which I intend to explore in the next few years.

1 Current Research

My research methodology combines a mix of theoretical analysis and system building to address practical data management problems. In this section, I will summarize a sample of my research proposals.

Denial Constraints Discovery. I have proposed the use of denial constraints (DCs) as a formal language to specify data quality rules in a declarative manner, and I have implemented the data quality rule discovery system RuleMiner to automatically discover candidate DCs from a sample (possibly dirty) data instance [6, 8]. Denial constraints (DCs) are a subset of first-order logic. They are able to capture many real-world data quality rules that previous integrity constraint formalisms, such as functional dependencies and check constraints, cannot express. For example, the data quality rule stating that “for two employees in the same role, the one based in NY cannot earn less than the one based in another state” can be expressed as a DC. RuleMiner employs a set of novel pruning rules to discover interesting (non-trivial and minimal) DCs. Since some of the discovered DCs might not be the correct, for example, due to overfitting the sample instance or errors in the data, they have to be verified by users. RuleMiner ranks the DCs according to a novel interestingness function, and shows every discovered DC with examples and an English description to assist users in the verification process. RuleMiner is adopted to assist users in defining data quality rules in NADEEF, the first open-source rule-based data cleaning system (available on Github [2]).

Distributed Data Deduplication. One major challenge faced by error detection techniques is the ability to scale to a large data set. I have addressed the scalability challenge in the context of data deduplication [4]. Data deduplication techniques usually require computing a similarity score for each pair of tuples. For a dataset with \( n \) tuples, naively comparing every tuple pair requires \( O(n^2) \) comparisons, a prohibitive cost when \( n \) is large. Leveraging shared-nothing data processing environments, which are widely adopted to store “big data”, I have designed a strategy to distribute the workload into a set of parallel machines with provable theoretical guarantees in terms of both the communication cost and the computation cost [4]. Though developed for detecting duplicate records, the proposed distribution strategy enjoys many other use cases. In fact, it can be used to distribute any workload where a self-join is necessary, for example, in detecting violations for DCs that involve two tuples and in validating an automatically discovered DC. It has been used for analyzing massive amounts of semi-structured RDF data [12], since the data triples in RDF data often need to be joined for analytical tasks.

Holistic Data Cleaning. Current data cleaning techniques usually address one type of errors at a time, which could lead to inaccurate repairs due to the lack of global context of errors. I have proposed the algorithm HolisticDataCleaning [7], which advocates the idea of accumulating evidence from various sources for detecting and repairing errors in a holistic manner to achieve better cleaning accuracy. HolisticDataCleaning is the core algorithm adopted in several open-source data cleaning systems, including NADEEF and Llunatic[3] HolisticDataCleaning has also been proven extremely valuable in detecting and repairing errors in real data in consultation with animal scientists at UC Berkeley studying the effects of firewood cutting on small terrestrial vertebrates (birds, amphibians, reptiles, and small mammals). I used the RuleMiner [6, 8] system to produce a set of DCs over this data set consisting of 60,000 records, with each record containing detailed information about each random capture of an animal. Many discovered DCs

2 https://github.com/Qatar-Computing-Research-Institute/NADEEF
3 https://github.com/donatellosantoro/Llunatic
were later confirmed by the animal scientists to be correct constraints. Given the confirmed DCs, I helped them to correct hundreds of errors using HolisticDataCleaning; those errors would otherwise be very difficult to spot by humans. The HolisticDataCleaning algorithm is also currently used in a project with Thomson Reuters to detect and repair errors in their data sets.

**Leveraging Knowledge Bases and Experts for Cleaning.** Automatic data repairing algorithms usually rely on the redundancy of the current data set and produce heuristic repairs or data updates. Therefore, a natural question is “can we produce reliable updates by leveraging external trustworthy data sources?” I have explored the use of knowledge bases (KBs) and human experts for repairing data in the KATARA system [9, 10]. KATARA first establishes the correspondence between the dirty database and the available KBs; the data errors are then detected and possible repairs are suggested according to the discovered correspondence. Since human experts are expensive, they are only involved when there is ambiguity in establishing the correspondence or when there is uncertainty in detecting data errors (due to the incompleteness of the KBs). As far as I know, KATARA is the first system that exploits the use of KBs and experts to repair data in a principled way. It is able to leverage existing general purpose knowledge bases such as Yago, DBPedia, and Freebase, as well as special purpose and enterprise curated KBs such as RxNorm. In addition, due to the use of KBs, the questions asked to human experts in KATARA are easily understandable (e.g., “is Madrid a capital?” or “does Spain hasCapital Madrid?”).

**2 Future Research**

The rapid increase of modern data-driven applications are rampant with data management challenges. I intend to continue working on solving practical data management problems through both theoretical analysis and system building.

**Managing IoT Data.** The data cleaning techniques I developed in my PhD thesis have mostly focused on relational data that are usually rich in semantics, for example, customer database and personnel records. Today, data is increasingly generated by sensors, mobile devices, and automated processes in a variety of domains, including manufacturing, transportation, healthcare, and smart homes, which are broadly referred to as the Internet of Things (IoT). These data are in various formats, and are filled with glitches or errors. Because the majority of this data reflects normal-case operation, it is generally not productive for analysts to exhaustively explore every value. Instead, a natural approach is to look for unusual values, or outliers. Surfacing anomalies and errors in IoT data is becoming increasingly important in ensuring a smooth functioning of these systems. Examples include equipment monitoring (e.g., detecting faulty manufacturing components), intrusion detection (e.g., building break-ins), environmental observation (e.g., occupancy and HVAC control), and customer service (e.g., finding customers experiencing poor service). Developing novel techniques and scalable systems to surface patterns, trends, and anomalies in the IoT data is one of my future research directions.

**Debugging Data Analytics Tools.** Abundant tools and commercial startups today exist for various kinds of data analytics. A common pitfall of these tools is that they often provide one-time analytics results. This is not an issue if the analytics results are acceptable; however, when the results are not desirable, users are often left with manual tuning without explanations of the results and without guidance for how parameters should be changed. Deep learning is one notorious example; it is very difficult for users to understand and to tune deep learning models. At the same time, it is usually not feasible to try all possible parameters. Therefore, a debugging facility could be useful in helping users navigate all possible configurations. I expect to leverage two essential technologies for developing a debugging facility. The first one is differential analysis, which correlates the difference between two analytics results to the difference between their input configurations. The second one is sampling with guarantees. Since running a tool for every configuration on the entire data might be expensive, sampling techniques are natural candidates for quickly testing a particular configuration on a sample.

**Extracting Value from Unstructured Data.** It is estimated that around 80% of potentially usable information is in unstructured formats such as emails, PDF documents, and Web pages. Extracting value
from unstructured data requires the extraction of high quality structured data, which can then be consumed by downstream analytics. While tools such as DeepDive exist for extracting structured data from unstructured data, measuring and ensuring the quality of the extracted data remains an important and challenging problem. The problem is challenging mainly because there is a gap between where the errors are detected and where the errors originated. On one hand, it is easier to spot errors in the extracted structured data than in the unstructured data, since many tools exist for detecting errors in structured data. On the other hand, the errors exist in the extracted data because either the extraction process is imperfect or the original unstructured data is incorrect. I envision developing tools to bridge the gap, and thus improving the quality of the extracted data.

References


