IJDC | *Conference Paper*

Conceptual Model and Framework for Collaborative Data Cleaning

Nikolaus Nova Parulian
University of Illinois at Urbana Champaign

Bertram Ludäscher
University of Illinois at Urbana Champaign

Introduction and Overview

Data cleaning and preparation are essential parts of data curation lifecycles and scientific workflow Ball [2012]. It is also known that exploratory data mining and data cleaning takes 80% of the scientific research pipeline Dasu and Johnson [2003]. However, a data cleaning task can be very tedious for a single user, involving lots of exploration and iteration, prone to error, especially when a curator finds various problems in the dataset. Nevertheless, the single-user data cleaning can also introduce bias where the cleaning quality will only be as good as their knowledge. Therefore, we can assign a data cleaning task to multiple data curators to collaborate on curating datasets. However, when a data cleaning task involves multiple users, it can introduce new problems such as data changes disagreement and conflicting process dependency. Understanding this variation on changes and analyzing the merging workflow is important for data curation to evolve the data cleaning workflow and improve the dataset's quality. In line with the reusability theme for IDCC 2022, this approach can help improve the data curation pipeline by improving the data cleaning pipeline through collaboration.

We can observe collaboration on data cleaning from different aspects, considering multiple possibilities or scenarios of how we want to clean the data. First, when the data cleaning tasks are clearly defined, we can look at collaboration to divide the tasks following the divide and conquer principle. The plan can be performed as a person can execute cleaning tasks on the independent columns (horizontal collaboration), independent rows (vertical collaboration), or independent tasks for different users based on the user expertise or specification. Second, collaboration can also be performed for redundancy purposes. This method seeks variation over the data cleaning steps or looks

Submitted Tuesday, Mar 15 2022

Correspondence should be addressed to

An earlier version of this paper was presented at International Digital Curation Conference.

The *International Journal of Digital Curation* is an international journal committed to scholarly excellence and dedicated to the advancement of digital curation across a wide range of sectors. The IJDC is published by the University of Edinburgh on behalf of the Digital Curation Centre. ISSN: 1746-8256. URL: http://www.ijdc.net/

at the agreement within different curators with a similar principle to the inter-annotator agreement. From a data curation perspective, this method allows the curator expert to preserve each dataset versioning, data cleaning recipes, or confidently choose the workflow that can satisfy the use of data the most (fitness for use).

Background and Motivation

Collaboration is always becoming an important topic to discuss when working with multiple people, skills, and capabilities. For example, collaborative software development has become a standard in software engineering to speed up the development lifecycle. Collaboration on the software development can be performed directly by dividing tasks into multiple functional units that will be integrated to be one usable application or library. There is also indirect collaboration which is popular in the open-source community where anyone can contribute to the project by fixing bugs or adding features to the existing version.

Single User Data Cleaning Model

In terms of the usability of the tools, software development can have different levels of reusability. First, **library development** which has high reusability because it can be useful for different applications with other developers as the target audience. Next, **application development** that targets the end-user as the audience makes a tool used by many people. Finally, a **"one-off" scripting** which is mainly developed for personal use, with a specific experiment/research as a target use case. It can also be shared for reproducibility or transparency purposes, but not necessarily for others to use.

The same idea can also be applied to data cleaning. For a single user, data cleaning workflow is similar to the "one-off" scripting where one can clean the dataset and be done with it. However, if we can generalize the data cleaning function into a column operation where dependencies can be analyzed, we can reuse the workflow by modularization Li et al. [2021], Parulian et al. [2021]. Furthermore, because the data cleaning itself is an iterative process, the data cleaning workflow itself can evolve where one can update the workflow if they find a new problem or bugs on the previous workflow or update the workflow for a new/updated dataset.

Collaborative Data Cleaning

We can apply the same principle to collaborative software development to a data cleaning task. The collaboration on data cleaning can be useful in many ways:

- We can speed up the cleaning process as in the divide and conquer principle. With multiple people working on different data segments simultaneously, the data cleaning tasks that require manual assessment can be done faster than a single-person task.
- Different people with different expertise can work on the data quality problem that they can tackle specifically. For example, one can focus on a missing value case, and the other can work on duplication, dependency constraint, or inconsistencies problems.

- Collaboration can also enable continuous development as the data cleaning task will be continued by another person for the same dataset or an updated dataset.

These benefits of collaboration can work because they have the same use-case or goal to clean the dataset so the data quality improvement within the result can be evaluated under the same metrics.

Approach and Challenges for Collaborative Data Cleaning

To apply the collaboration framework on data cleaning, we come up with three different approaches to dividing tasks:

- division based on independent column (horizontal collaboration): We assume two curators clean a dataset for different data cleaning use cases based on the problematic columns. Thus, the tasks will be divided based on columns' separation and possible dependency to minimize conflicting workflow. In the end, the collaboration framework should generate a way to merge the two data cleaning results.

- division based on independent row (vertical collaboration): In this type of collaboration, we assume two curators clean a dataset for the same set of data cleaning use cases with different data rows based on specific criteria. For example, on an employee table with a problematic address, we divide data cleaning tasks based on the employee's state. Since they only have access to their part, the cleaning results are relative to their work. Thus, besides merging the dataset, we also need to merge the workflow to make sure we have the combined workflow that can clean the overall dataset holistically.

- redundant cleaning: In this collaboration, we assume two curators will perform data cleaning for the same dataset with the same use case independently from each other. In this case, we can consider data cleaning as an annotation task that needs further analysis for the decision-making process. Therefore, the collaboration framework for this case should focus on reporting agreement or conflicting data (cell) and workflow that need resolution.

Problem: Duplicate ID

id	name	birth_date
1	John	Aug, 1 1988
1	Doe	1-Aug-1986
2	Alex	20-Jan-1993
3	Patricia	Feb 11, 1990

Curator A: Keep First

id	name	birth_date
1	John	Aug, 1 1988
2	Alex	20-Jan-1993
3	Patricia	Feb 11, 1990

Curator B: Keep Last

id	name	birth_date
1	Doe	1-Aug-1986
2	Alex	20-Jan-1993
3	Patricia	Feb 11, 1990

Curator C: Keep Both

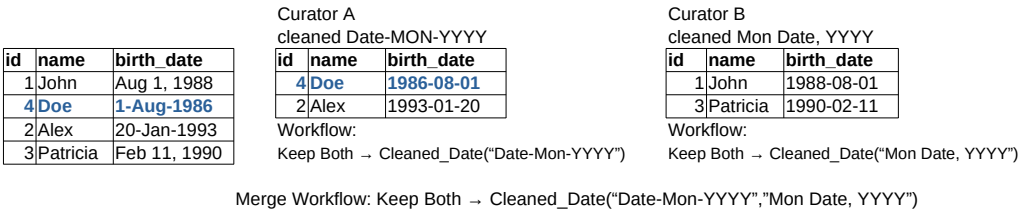
id	name	birth_date
1	John	Aug, 1 1988
4	Doe	1-Aug-1986
2	Alex	20-Jan-1993
3	Patricia	Feb 11, 1990

Figure 1. Direct Conflicts: Consider an employee dataset employee(employee_id, employee_name). This dataset has an inconsistencies problem where two ids have a different name employee(1, john) and employee(1, doe). Curator A fixed the dataset by admitting "john" as the truth value. Curator B fixed the dataset by using "doe" as the truth value. Whereas curator C considers both of them to be different entities, thus instead of choosing one, she updated the id to a new identifier, making both entities of their own.

doi:10.2218/ijdc.v0i0.0

As it suggested, merging the cleaned data and workflow from the collaboration and division strategy above can result in conflicting data changes or workflow dependency as follow:

- direct conflicts: merging results can induce direct conflicts (merge data conflicts). For example, as seen in Figure 1, a redundant cleaning strategy can result in direct conflicts where different curators clean the dataset differently. When these data are merged, there will be conflict on the resulting dataset that needs reporting and resolution.



id	name	birth_date
1	John	Aug 1, 1988
4	Doe	1-Aug-1986
2	Alex	20-Jan-1993
3	Patricia	Feb 11, 1990

id	name	birth_date
4	Doe	1986-08-01
2	Alex	1993-01-20

id	name	birth_date
1	John	1988-08-01
3	Patricia	1990-02-11

Figure 2. Indirect Conflicts: Supposed we choose to keep both for the duplicate IDs problem, and now we continue fixing the date. We want the birth_date to follow the date formatting "YYYY-MM-DD". However, the birth_date currently contains formatting inconsistencies with two standards (a) "Mon, Date, YYYY" and (b)"DD-Mon-YYYY". We split the task by the rows so Curator A can fix the problem (a) and Curator B for problem (b). Merging the result might not produce conflicts because they are working on independent rows (dataset). However, suppose we want to reuse the workflow for a whole/updated dataset. In that case, we need to consider the function dependency since executing the workflow sequentially (without division) will produce an inconsistent result.

- indirect conflicts: Two workflows can be merged without conflict, but reusing this merged workflow will fail the target use-case testing or produce a semantic error because they need additional filtering or missing parameters. For example, following the Figure 2 we need to analyze the dependency of the two different workflows from vertical collaboration because re-executing these data cleaning processes as it is without division or combined parameters will produce an inconsistent result.

Methods

On our previous work Parulian et al. [2020] we have developed DCM, a data cleaning provenance model that can capture provenance from a data cleaning task in a granularity of changes on cell, column, row, and schema. It can also capture workflow or recipe associated with the task, and with addition of column-level operation we can generalize the recipe to represent a workflow provenance model. Because the existing DCM capture existing provenance model from a sequence data cleaning operation, there is a potential on using the model for data cleaning collaboration purpose to perform data merging and process merging. One solution is data merging where we can merge, branch, or resolve conflicts on every cell change from two data cleaning tasks based on the dataset snapshot. Besides looking at the data merging, we will also analyze the workflow level

collaboration to understand the difficulties and resolution on merging the data cleaning workflow. Merging the data cleaning workflow will have its challenges because it requires analysis of the workflow dependency.

Tools and applications have been developed to support collaboration on the data cleaning task. For example, online spreadsheets such as *Microsoft Excel* Microsoft [2022] and *Google Sheets* Google [2022] allows a user to share a dataset file with different users, thus allowing multiple users to change the dataset synchronously. Although these online spreadsheet tools are convenient for document editing because we can see the changes right away, they do not preserve any form of provenance that can make the operation reusable. For specific data cleaning or wrangling purpose, proprietary tools such as Trifacta Data Wrangler Trifacta [2022] allow users to share or assign a data cleaning project to multiple users. From an open-source side, DataHub Bhardwaj et al. [2014] and CoClean Musleh et al. [2020] also has been developed to allow collaboration on cleaning tasks using centralized data processing. Compared to this previous work which focuses on application development, we want to see the collaboration from a conceptual model perspective by presenting a framework to use existing provenance artifacts from multiple users and different data cleaning scenarios for collaboration. In addition to the existing provenance model DCM, we want to provide a transparent, collaborative, technology-independent data cleaning framework that we can apply to the existing data cleaning tool Figure 3.

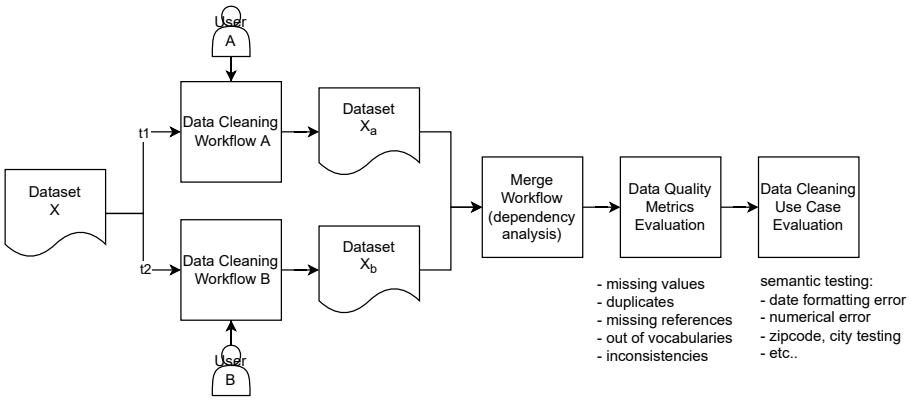


Figure 3. Collaboration Framework: when a data cleaning plan has been declared, two curators work on cleaning the dataset and produce different data cleaning workflow with provenance information. We merged the workflow and analyzed the process dependency to minimize conflict and report required resolution. The possible merged plan will be evaluated for syntactic data quality metrics and the use cases based on the test case.

With this research on using provenance information for collaborative data cleaning, we want to deliver these contributions

- conceptual provenance model and alignment framework to support indirect collaboration.
- provenance analysis for merging cleaned dataset, reporting agreements, conflicts, and possible resolution.
- prototypical implementation of the collaboration workflow on existing data cleaning tool (OpenRefine).

We hope this conceptual model can be a one-step toward transparent and collaborative data cleaning.

References

Alex Ball. Review of Data Management Lifecycle Models. *University of Bath*, page 15, 2012.

Anant Bhardwaj, Souvik Bhattacherjee, Amit Chavan, Amol Deshpande, Aaron J. Elmore, Samuel Madden, and Aditya G. Parameswaran. DataHub: Collaborative Data Science & Dataset Version Management at Scale. *arXiv:1409.0798 [cs]*, September 2014. URL http://arxiv.org/abs/1409.0798. arXiv: 1409.0798.

Tamraparni Dasu and Theodore Johnson. *Exploratory data mining and data cleaning*, volume 479. John Wiley & Sons, 2003.

Google. Google Sheets: Free Online Spreadsheet Editor | Google Workspace, 2022. URL https://www.facebook.com/GoogleDocs/.

Lan Li, Nikolaus Parulian, and Bertram Ludäscher. Automatic Module Detection in Data Cleaning Workflows: Enabling Transparency and Recipe Reuse. In *16th International Digital Curation Conference (IDCC)*, 2021. doi: 10.5281/zenodo.5606219. https://doi.org/10.5281/zenodo.5606219.

Microsoft. Microsoft Excel Spreadsheet Software | Microsoft 365, 2022. URL https://www.microsoft.com/en-us/microsoft-365/excel.

Mashaal Musleh, Mourad Ouzzani, Nan Tang, and AnHai Doan. CoClean: Collaborative Data Cleaning. In *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*, pages 2757–2760, Portland OR USA, June 2020. ACM. ISBN 978-1-4503-6735-6. doi: 10.1145/3318464.3384698. URL https://dl.acm.org/doi/10.1145/3318464.3384698.

Nikolaus Nova Parulian, Timothy M McPhillips, and Bertram Ludäscher. A model and system for querying provenance from data cleaning workflows. *Provenance and Annotation of Data and Processes*, pages 183–197, 2020. URL http://dx.doi.org/10.1007/978-3-030-80960-7_11.

Nikolaus Nova Parulian, Lan Li, and Bertram Ludaescher. or2yw: Modeling and Visualizing OpenRefineHistories as YesWorkflow Diagrams. *arXiv:2112.08259 [cs]*, December 2021. URL http://arxiv.org/abs/2112.08259. arXiv: 2112.08259.

Trifacta. The Trifacta Data Engineering Cloud, 2022. URL https://www.trifacta.com.