C²Metadata: Automating the Capture of Data Transformations from Statistical Scripts in Data Documentation

Jie Song
University of Michigan
Ann Arbor, Michigan
jiesongk@umich.edu

George Alter
University of Michigan
Ann Arbor, Michigan
altergc@umich.edu

H. V. Jagadish
University of Michigan
Ann Arbor, Michigan
jag@umich.edu

ABSTRACT

Datasets are often derived by manipulating raw data with statistical software packages. The derivation of a dataset must be recorded in terms of both the raw input and the manipulations applied to it. Statistics packages typically provide limited help in documenting provenance for the resulting derived data. At best, the operations performed by the statistical package are described in a script. Disparate representations make these scripts hard to understand for users. To address these challenges, we created Continuous Capture of Metadata (C²Metadata), a system to capture data transformations in scripts for statistical packages and represent it as metadata in a standard format that is easy to understand. We do so by devising a Structured Data Transformation Algebra (SDTA), which uses a small set of algebraic operators to express a large fraction of data manipulation performed in practice. We then implement SDTA, inspired by relational algebra, in a data transformation specification language we call SDTL.

In this demonstration, we showcase C²Metadata's capture of data transformations from a pool of sample transformation scripts in at least two languages: SPSS®and Stata®(SAS®and R are under development), for social science data in a large academic repository. We will allow the audience to explore C²Metadata using a web-based interface, visualize the intermediate steps and trace the provenance and changes of data at different levels for better understanding of the process.

SIGMOD '19, June 30-July 5, 2019, Amsterdam, Netherlands

ACM ISBN 978-1-4503-5643-5/19/06…$15.00
https://doi.org/10.1145/3299869.3320241

CCS CONCEPTS

• **Information systems** → **Data provenance**; **Extraction, transformation and loading**.

KEYWORDS

data transformation, data documentation, data provenance

ACM Reference Format:
Jie Song, George Alter, and H. V. Jagadish. 2019. C²Metadata: Automating the Capture of Data Transformations from Statistical Scripts in Data Documentation. In *2019 International Conference on Management of Data (SIGMOD '19), June 30-July 5, 2019, Amsterdam, Netherlands.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3299869.3320241

1 INTRODUCTION

As the research community responds to increasing demands for public access to scientific data, the need for improvement in data documentation has become critical. Accurate and complete metadata is essential for data sharing and for interoperability [3]. However, the process of describing and documenting scientific data has remained a tedious, manual process even when data collection is fully automated.

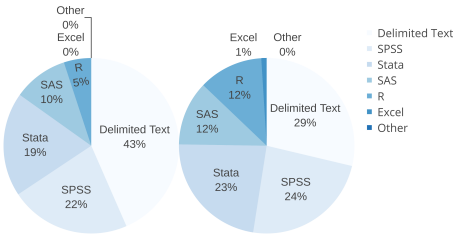


Figure 1: ICPSR data downloads by format (Sep. 4, 2015 - Mar. 4, 2016)

Researchers in many fields use the main statistics packages for data management as well as analysis. Figure 1 shows data downloads by statistical pakcage format, mainly SPSS®, SAS®, Stata®and R [4, 6–8], from the Inter-university Consortium for Political and Social Research (ICPSR) [1]. These

[1]https://www.icpsr.umich.edu/

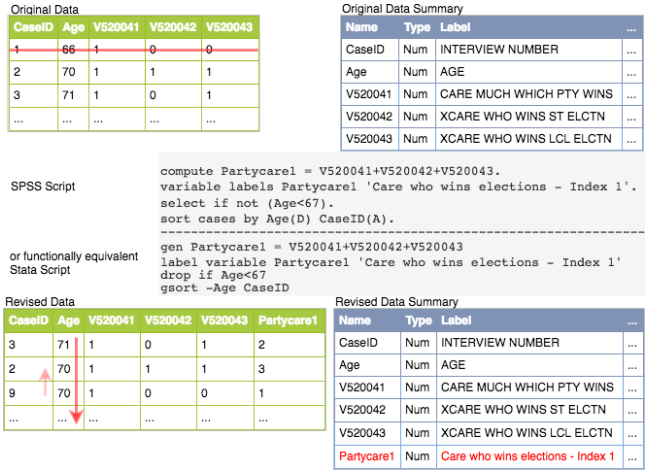


Figure 2: Data Transformation Example by SPSS and Stata scripts

packages, however, lack tools for documenting variable transformations in the manner of a workflow system or even a database. At best, the operations performed by the statistical package are described in a script, which more often than not is not even available to future data users. Different statistics packages differ in data model, transformation representation and scope of transformations covered; thereby further complicating the understanding of the transformation process.

Motivating Example. *Figure 2 is an example of data analysis using statistical languages. The original data is part of a computer-generated survey result of political opinions. It records surveyed attributes: CaseID uniquely identifies survey participants while V520041 to V520043 represent whether the participant cares about who wins party, state or local election. As the user is interested in all those who care about who wins elections, she generates a transformation script that computes the union of opinions by adding V520031 to V520043 as a newly labeled attribute Partycare1. She then cleans the data by filtering out cases whose Age is less than 67 before sorting cases in the descending order of Age, breaking ties by the ascending CaseID. Here we show functionally equivalent transformation scripts in two languages: SPSS and Stata. For more procedural languages like R, transformations can be even harder to interpret without familiarity of the language.*

To reduce the cost and increase the completeness of metadata, we aim to work with common statistical packages to automate the capture of metadata at the granularity of individual data transformations in a simple yet expressive representation regardless of the original languages used. In this demo, we show C^{2}Metadata as such a system that implements this idea, creating efficiencies and reduce the costs of data collection, preparation, and re-use. The system first

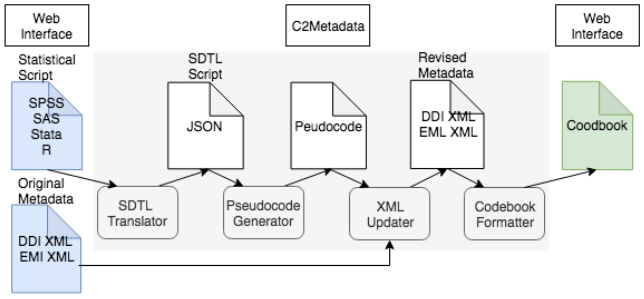


Figure 3: C^{2}Metadata workflow

reads statistical transformation scripts (in SPSS, Stata, SAS or R) and the original metadata in (either of) two internationally accepted XML-based standards: the Data Documentation Initiative (DDI) [9] and Ecological Metadata Language (EML) [2]. It then walks over individual data transformations and uses a software-independent data transformation description (Standard Data Transformation Language (SDTL)) to update the original metadata, which permits the tracking of dataset changes at different levels. We define SDTL based on a small set of transformation operators that comprise the Standard Data Transformation Algebra (SDTA), and cover the majority of data transformations used in statistical software. SDTL is a declarative language describing the purpose of transformations in an informative way. To ease the understanding of the process without the necessity of learning SDTL or SDTA, we further extend the interpretation from SDTL to natural language as part of the updated metadata. We currently target two research communities (social and behavioral sciences and earth observation sciences) with strong metadata standards that rely heavily on statistical analysis software. However, we believe our work is generalizable to other domains, such as biomedical research. More details of the project is available at http://c2metadata.org.

Next, we provide a brief overview of C^{2}Metadata and SDTL and then an overview of the actual demonstration.

2 C^{2}METADATA OVERVIEW

In this section, we discuss the pipelined modular architecture of C^{2}Metadata comprising four modules, as shown in the grey area in Figure 3. We explain the system workflow and then describe SDTA and SDTL.

2.1 Workflow

The workflow in C^{2}Metadata consists of four modules: SDTL Parser, Pseudo-code Generator, XML Updater and Codebook Formatter.

SDTL Parser. In the Script Parser module, C^{2}Metadata expresses data transformations in SDTL, independent of the

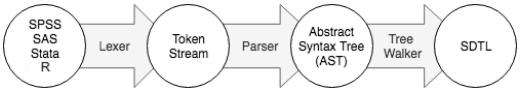
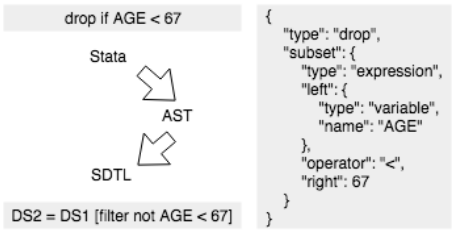


Figure 4: SDTL Parser Components

language used. The components of the module are shown in Figure 4.

Script Parser is customized for each statistical package since they each use a different scripting language. It takes a command script written in scripting languages as input and uses standard compiler techniques to parse input, obtain a syntax tree, and then generate SDTL code. The syntax Lexer transforms the raw SPSS/Stata/SAS/R syntax code by a lexer grammer for each language into a stream of tokens. The ANTLR-based [5] Parser then parses the token stream into an abstract syntax tree (AST), simple or nested. The Tree Walker finally walks over the AST and refers to a mapping between statistical languages and SDTL for SDTL translation. Below is the translation of an example of a simple command in Stata by Stata Script Parser.



Since R is a more dynamic and open language than SPSS, Stata or SAS, we limit our scope of translation to the set of transformation-based functions in the base and tidyverse [10] packages.

The SDTL Translator issues an error when illegal or unrecognized commands are found in inputs or intermediate steps.

Pseudocode Generator. The translated SDTL script is then converted to human readable text for a more user-friendly illustration of the transformations included.

XML Updater. The original metadata in DDI or EML standards are updated with both file level and data element level transformations including the original transformation script, the natural language description of transformations and the SDTL equivalent in XML format. (DDI and EML are both based on XML).

Codebook Formatter. An HTML codebook is generated from the revised metadata describing the contents, structure and layout of the revised data.

C²Metadata is deployed as a Docker container since multiple collaborators contribute to the development of the project using technologies including .NET, closure, Java, XSLT, COGS, among others.

Table 1: Operations supported by SDTL

dataset level	load, save, rename, create, merge, match, transpose, format display
column/variable level	recode, rename, aggregate, compute, delete, label, sort, missing value, join
row/case level	select, sort, add, delete, aggregate
cell level	update, label
procedural	if-then, do repeat loop

2.2 SDTA and SDTL

Inspired by Relational Algebra, we define Standard Data Transformation Algebra (SDTA) as a realization of statistical data transformation using a small set of primitive operators. This permits us to write complex transformations as a single SDTA expression, and to optimize evaluation by using algebraic equivalences between SDTA expressions. We do not have the space to describe the algebra in full here, but we will show examples in the demo.

Just like SQL build on relational algebra, we define SDTL as an extensible declarative language that can express all of SDTA using Convention-based Ontology Generation System (COGS) [1] information model. SDTL offers multiple representations such as XML, JSON, GraphQL under one specification. It supports the most basic and widely used data transformation operations in the four main statistical languages from major data production projects (the General Social Survey (GSS)[2], the American National Election Study (ANES)[3], and the National Survey of Family Growth (NSFG)[4]), ICPSR, DataOne[5] and sample scripts provided to journals in conjunction with replication datasets. Each operation is bundled with a natural language interpretation template to better illustrate the associated operation. Table 1 summarizes the operations currently supported by SDTL.

3 DEMONSTRATION SCENARIO

We will present an end-to-end demonstration of C²Metadata capturing the data transformation in statistical scripts as part of the metadata using an interactive web interface (Figure 5). We also allow the user to interact with the system to visualize intermediate results and track the provenance of data at different levels with more social science data and metadata transformation examples. We will perform the demo on a transformation script in SPSS/Stata and a DDI 2.5 XML file

[2]http://gss.norc.org/
[3]https://electionstudies.org/
[4]https://www.cdc.gov/nchs/nsfg/index.htm
[5]https://www.dataone.org/

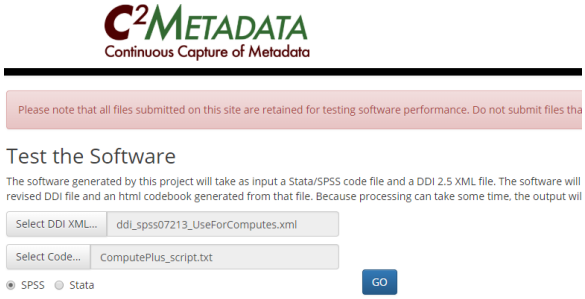
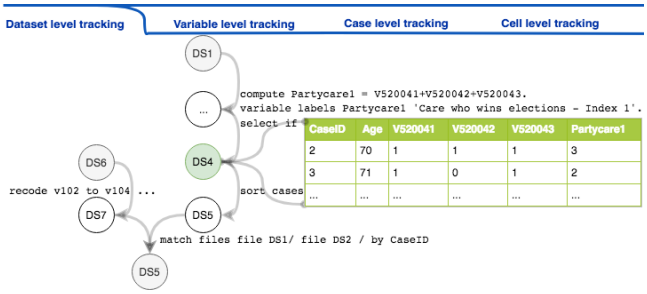


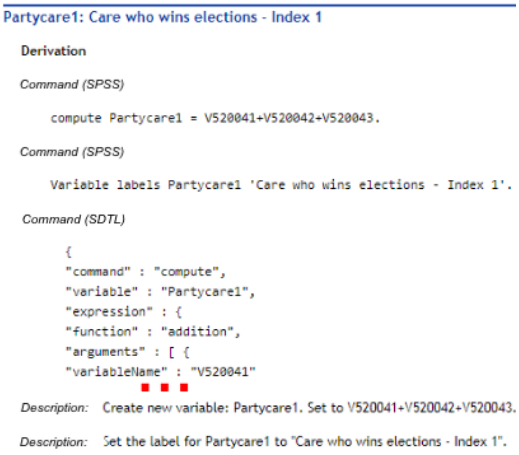
Figure 5: C²Metadata Upload Interface

from a pool of samples in the social science domain, obtained from ICPSR. We upload the two files at the Upload Interface and select the statistical language of the transformation script. The DDI file is then updated with the transformation described in the scripts in SDTL. It eventually returns the revised DDI file and an html codebook generated from the file.

Apart from the revised metadata, at the Transformation Tracking Interface, C²Metadata allows tracking of the processed transformations at four different levels: dataset level, variable level, case level and cell level, in multiple settings and transformation representations. Figure 6a shows the tracking of the transformation of the sample at the dataset level in a graphical setting, where each node is a dataset after one step of transformation. Here we present the transformation by the original SPSS representation. By clicking on the node, the intermediate transformation result will expand for inspection. Variable-, case-level and cell-level give more provenance information of the object of interest at the current stage or along the transformation trace. An example of the tracking of variable Partycare1 is shown in Figure 6b in a codebook setting. In this senario, The transformations applied specifically to this variable are represented by the original language (SPSS), SDTL and a human-readable natural language description.



(a) Graphical visualization at dataset level



(b) Cookbook setting at variable level

Figure 6: Transformation Tracking Examples

4 ACKNOWLEDGMENTS

This research is supported in part by the National Science Foundation through grant ACI-1640575 and IIS-1250880. For the development of the system, we thank Colectica and Norwegian Centre for Research Data (NSD) for the specification of SDTL, Colectica for SPSS Parser, NSD for Stata Parser, Yashas Vaidya for SAS Parser, Metadata Technology North America (MTNA) for DDI Updater and ICPSR for Pseudocode Generator and Codebook Formatter.

REFERENCES

[1] Colectica. 2017. Convention-based Ontology Generation System (COGS) 1.0. http://cogsdata.org/docs/.

[2] Eric H Fegraus, Sandy Andelman, Matthew B Jones, and Mark Schildhauer. 2005. Maximizing the value of ecological data with structured metadata: an introduction to Ecological Metadata Language (EML) and principles for metadata creation. *The Bulletin of the Ecological Society of America* 86, 3 (2005), 158–168.

[3] Boris Glavic and Klaus R Dittrich. 2007. Data Provenance: A Categorization of Existing Approaches.. In *BTW*, Vol. 7. 227–241.

[4] SAS Institute. 2015. SAS®9.4 Product Documentation. http://support.sas.com/documentation/94/index.html

[5] Terence J. Parr and Russell W. Quong. 1995. ANTLR: A predicated-LL (k) parser generator. *Software: Practice and Experience* 25, 7 (1995), 789–810.

[6] R Core Team. 2013. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. http://www.R-project.org/

[7] IBM SPSS et al. 2017. IBM SPSS Statistics for windows, version 25.0. *IBM Corp., Armonk, NY* (2017).

[8] A Stata, Press Publication, and Statacorp Lp. 2015. Stata Base Reference Manual Release 14.

[9] Mary Vardigan, Pascal Heus, and Wendy Thomas. 2008. Data documentation initiative: Toward a standard for the social sciences. *International Journal of Digital Curation* 3, 1 (2008).

[10] Hadley Wickham. 2017. *tidyverse: Easily Install and Load the 'Tidyverse'*. https://CRAN.R-project.org/package=tidyverse R package version 1.2.1.