International Journal of Data Warehousing and Mining
Volume 15 • Issue 4 • October-December 2019

A Comparative Study of Data Cleaning Tools

Samson Oni, University of Maryland Baltimore County, USA
Zhiyuan Chen, University of Maryland Baltimore County, USA
Susan Hoban, University of Maryland, Baltimore County, USA
Onimi Jademi, University of Maryland, Baltimore County, USA

ABSTRACT

In the information era, data is crucial in decision making. Most data sets contain impurities that need to be weeded out before any meaningful decision can be made from the data. Hence, data cleaning is essential and often takes more than 80 percent of time and resources of the data analyst. Adequate tools and techniques must be used for data cleaning. There exist a lot of data cleaning tools but it is unclear how to choose them in various situations. This research aims at helping researchers and organizations choose the right tools for data cleaning. This article conducts a comparative study of four commonly used data cleaning tools on two real data sets and answers the research question of which tool will be useful based on different scenario.

KEYWORDS

Big Data, Data Cleaning, Data Cleansing, Data Fusion, Data Quality, Data Wrangler, Dirty Data, Open Refine

INTRODUCTION

Data is constantly being produced in every sector. However, data is produced in many forms, with various levels of quality and some data may have poor quality. Data cleaning, sometimes called data scrubbing or data cleansing, is the detection and removal of errors and inconsistency from data with the aim of improving data quality. In Big Data processing, data cleaning is a critical and important step prior to data processing and maintenance (Müller & Freytag, 2005). Data cleaning is important to both data from a single source and data from multiple sources. Data cleaning is an essential step for the data fusion process, which is the process of merging data from multiple sources (Haghighat, Abdel-Mottaleb, & Alhalabi, 2016). Fusing poor quality data from various sources together will cause more issues afterwards. Therefore, adequate cleaning of data from various sources before integration will have significant impact on the outcome of data fusion.

Cleaning data requires identifying incorrect, invalid or duplicate entries. The quality of data is determined by the degree to which the data in question meets specific needs, which in any case will be higher as the data becomes cleaner (Kandel, Paepcke, Hellerstein, & Heer, 2011). Validity, completeness, accuracy and precision are the measures of data quality (Kandel et al., 2011). The importance of accurate and correct data for fusion/ETL process cannot be over emphasized.

DOI: 10.4018/IJDWM.2019100103

Data analysts also spend a great deal of time and resources trying to fix data quality problems. Dasu et al. (Dasu & Johnson, 2003) emphasized the rule of thumb, which states that more than eighty percent (80%) of time on a data analysis project is spent on cleaning and preprocessing.

Although there are many data cleaning tools, they often have distinctive features. They also require distinct levels of skills to use them and have different costs and learning curves. Determining the best tools for any given cleaning task depends on many factors. However, in practice, users are often not experts on data cleaning tools and technologies so there is great need to provide some guidance on how to choose data cleaning tools.

The objective of this paper is to analyze four popular data cleaning tools and determine which tools are appropriate for various scenarios. This paper compares the features of these tools and their performance on cleaning the same dataset. Two data sets were used for this experiment. The results may help users choose appropriate data cleaning tools.

This paper makes the following contributions:

- Compared the performance of four data cleaning tools on two real world data sets. The metrics include their features, required platforms and skill level, time of completion, ease of implementation/usage, etc.
- Proposes a guideline for choosing data cleaning tools.

The rest of the paper is organized as follows. A background study is presented first, followed by an overview of various aspects of data cleaning. The methodology section describes the methodology used for the comparison study. The results section describes the results of the study. The discussion and conclusion section present the guidelines for choosing data cleaning tools and concludes the paper.

LITERATURE REVIEW

There has been lot of work on data cleaning. The work can be roughly divided into two categories: those on methods to address specific data quality issues and those on more general tools or framework that can address multiple data quality issues.

Work on specific data quality issues: Lee et al. (Lee, Lu, Ling, & Ko, 1999) presented several techniques to preprocess records before sorting them so that potentially matching records will be brought to close together. Using these techniques, they implemented a data cleaning system that can detect and remove duplicate records.

Various methods of handling missing data were discussed by Luján-Mora (Martinez-Mosquera et al., 2017). The authors proposed algorithms used in an analysis of an incomplete data set. The authors proposed multiple imputation methods, including regression imputation (filling in missing data with values predicted by a regression model) and single hot deck imputation (replacing the missing values with those obtained from similar objects from the same experiments).

General tools or framework: Martinez-Mosquera et al. (Martinez-Mosquera, Luján-Mora, López, & Santos, 2017) looked at modeling data cleaning for Big Data analysis based on previous research for modeling ETL processes using what is known as Unified Modeling Language (UML). They presented two use cases, one for modeling the data cleaning process for web logs and the other for modeling the cleaning process for security logs.

Galhardas et al. (Galhardas, Florescu, Shasha, & Simon., 2000) developed a data cleaning framework called AJAX. Their approach separates physical and logical levels of data cleaning. The logical level supports the design of the data cleaning workflow and the physical level implements the data cleaning workflow. This framework transforms existing data from one or more data collections to a target schema while eliminating duplicates.

Luján-Mora et al. (Martinez-Mosquera et al., 2017) proposed a technique for data cleaning that can be used for checking data quality issues on security logs. They used predefined rules to combine log data and scan for issues. Detected issues are then corrected before the data set is analyzed. However, their work focused on a specific data type: security logs.

Kandel et al. (Kandel et al., 2011) described Data Wrangler as a tool for the interactive cleaning of data using visual specifications of data transformation scripts. They explained that Data Wrangler combines direct manipulation of visualized data with automatic inference of relevant transformations which in turn cleanse the data.

Karrar and Ali (Karrar & Ali, 2016) conducted a comparative analysis of SQLServer and Winpure tools using academic and weather datasets. They analyzed two data cleaning tools, while our approach used four tools that are more commonly used in the industry for data cleaning. Porwal & Vora (Porwal & Vora, 2013) also carried out comparative analysis on two data cleaning algorithms: The Alliance rule algorithm and Hadclean algorithm. However, they did not compare more full-fledged data cleaning tools. They also did not consider other factors such as usability of the tools.

In a nutshell, there has been a lot of research on the need for data cleaning as well as data cleaning techniques, tools, and frameworks. However, there is not much work on comparing tools and choosing data cleaning tools and the criteria to consider. This paper conducts a comparative study on four commonly used data cleaning tools.

OVERVIEW OF DATA CLEANING

This section will describe types of data, data quality issues considered in this paper, general steps of data cleaning for a single source, and general steps of data cleaning for multiple sources.

Types of Data

With respect to categorization of data, there are three types of data: structured, semi-structured and unstructured data.

1. **Structured data:** This type of data has a high degree of organization and adheres to a predefined data model. One example is data in a relational database.
2. **Semi-structured data:** This type of data does not fit into relational database but have some form of organization for easy analysis. An example is XML data.
3. **Unstructured data:** This data type is not organized nor does it have a predefined model. Unstructured data is not a good fit for relational database. Examples are text, pdf, images.

Irrespective of the type of data, poor data quality will lead to poor analysis and decisions.

Data Quality Issues

Data quality issues can come in various forms ranging from duplicate data, missing data, errors (like spelling student as stdent), to inconsistent format etc. Below are several types of data quality issues considered in this paper.

1. **Misspelled data:** For example, a column has ‘student’, another has ‘stdent’ which is misspelled.
2. **Duplicate data/records (Table 1):** This is when the same information is entered or duplicated in a dataset or database.
3. **Irrelevant data**: this is the data in the dataset that are not relevant to the work. This kind of data needs to be removed.
4. **Mixed ranges:** Sometimes data is measured in ranges, e.g., salary, age. Ranges need to be represented consistently and appropriately in the data.

Table 1. An example of Duplicate record

No	First Name	Last Name	Age	Sex	Phone No
1	Samson	James	19	M	202-298-2014
2	Samson	James	19	M	202-298-2014

5. **Mixed numerical scales:** this type of data deals with using different numerical scales of data in the dataset. For example, representing one million can be represented as 1m while one billion can be represented as 1B. But a computer may not easily get this.
6. **Multiple representation:** representing the same piece of information in different forms yet having the same meaning can cause problems within a data set. For example, using multiple representations for the country United States (i.e. U.S. A, US, United States, United States of America). All these representations mean the same but using a mixture of several representations for the same information within a data set will cause trouble for analysis.
7. **Wrong date format:** Different date format are used in data today, but the mixture of several data formats in one dataset can be troublesome. Example of different formats can be 2/12/2018, February 2, 2018, and 2-12-2018. The three dates mean the same, but their presentation differs. Another example of date inconsistency is the American (MM/DD/YYYY) and European (DD/MM/YYYY) formats mixture. In American format the day will be written as 2/12/2018 to be 12th of February 2018, while Europeans will represent the same date as 12/2/2018 starting with the day.

General Steps of Data Cleaning for a Single Source

There are several phases involved in data cleaning for a single data source.

- Detect errors and inconsistencies in data to remove.
- Verify that the error is really an error, not a special feature of the dataset (Rahm & Do, 2000). This often requires human interaction.
- Extract erroneous records to a new temporary table.
- Perform cleaning operations on the data in that temporary table.

Most of these processes are already built in different data cleaning tools as doing this manually will cost lot of time and resources.

General Steps of Data Cleaning for Multiple Sources

In multiple data sources, each data source may contain dirty data. In addition, data from one source may contradict or overlap with data from other sources. The process of merging these data is also known as data fusion.

Table 2 depicts a single data source that has a few data quality issues including misspelling (Nigeria was spelled as Nigria) and duplicated data.

Table 3 and Table 4 show two data sources that need to be integrated. Each data source may have some data quality issues (e.g., Nigeria is misspelled in source 2). Some of these issues can be addressed in the individual source, but others can only be addressed during and after integration.

Table 5 shows integrated data in which cleaning was done in the individual source alone. Issues like misspelling are addressed, but redundancy and overlapping are not. For example, we have columns for Name, firstname and lastname, and columns for sex and gender. Table 6 shows clean integrated data where these issues are addressed.

Table 2. Data Quality issues in a single data source.

CampusId	First Name	Last Name	Country	Sex
VH609042	Samson	James	Nigria	M
XV503267	Jane	Mark	India	F
XV503267	Jane	Mark	India	F

Table 3. Data in source 1

CampusId	Name	Address	Sex	Date of Birth
VH609042	Samson James	100 IRC 21222, MD	1	12-01-1989
XV503267	Jane Mark	123 Ocean street, 21223, MD	0	02-01-1988

Table 4. Data in Source 2

CampusId	First Name	Last Name	Country	Gender	Course
VH609042	Samson	James	Nigria	M	10
XV503267	Jane	Mark		F	10

Table 5. Integrated data with data cleaning in individual sources only

CampusId	Name	Address	Date of Birth	Sex	First Name	Last Name	Country	Gender	Courses
VH609042	Samson James	100 IRC 21222, MD	12-01-1989	1	Samson	James	Nigeria	M	10
XV503267	Jane Mark	123 Ocean street, 21223, MD	02-01-1988	0	F	Mark	India	F	10

Table 6. Clean Integrated data

CampusId	FirstName	LastName	Sex	Address	Courses	Date of Birth	Country
VH609042	Samson	James	M	100 IRC 21222, MD	10	12-01-1989	Nigeria
XV503267	Jane	Mark	F	123 Ocean street, 21223, MD	10	02-01-1988	India

Aside from the issue of data overlapping associated with multiple sources, naming and structural conflicts may occur (Batini, Lenzerini, & Navathe, 1986) (Parent & Spaccapietra, 1998). Naming conflicts is an issue that arises when different names are used for same objects across sources, or when the same name is used for different objects across sources. Meanwhile, structural conflicts occur when different representations of the same object arise in different data sources.

So the general steps to clean data from multiple sources include 1) clean data at each source; 2) data integration; 3) address data quality issues in integrated data.

METHODOLOGY

This section describes the methodology of the comparative study, including the data sets, the four data cleaning tools, and the data cleaning tasks.

Data Sets

In this work, we used a data set on atmospheric and climate research from the U.S Department of Energy website (www.arm.gov) and a data set about universities (universityData) extracted from Wikipedia. The data from the U.S Department of Energy website is the Atmospheric Radiation Measurement (ARM) user facility data collected through scientific experiments and routine operations. The observations were made every half an hour. The University data set gives an overview of different universities: when they were established, the number of faculty, staff and students currently enrolled as well as the total endowment amount each university currently possesses. The information included in the dataset is explained in Table 7.

The University dataset has 10 variables (p=10), contains over 75,000 records (n= 75043) and is saved as CSV. The ARM dataset has 15 variables (p=15), contains over 12,000 records (n= 12,762) and is saved as CSV. Table 8 shows the columns in university data.

Data Quality Issues in Data Sets

Figure 1 and Figure 2 show the screen shots of these two data sets, respectively. The two data sets used for experiments are very messy and have several data quality issues:

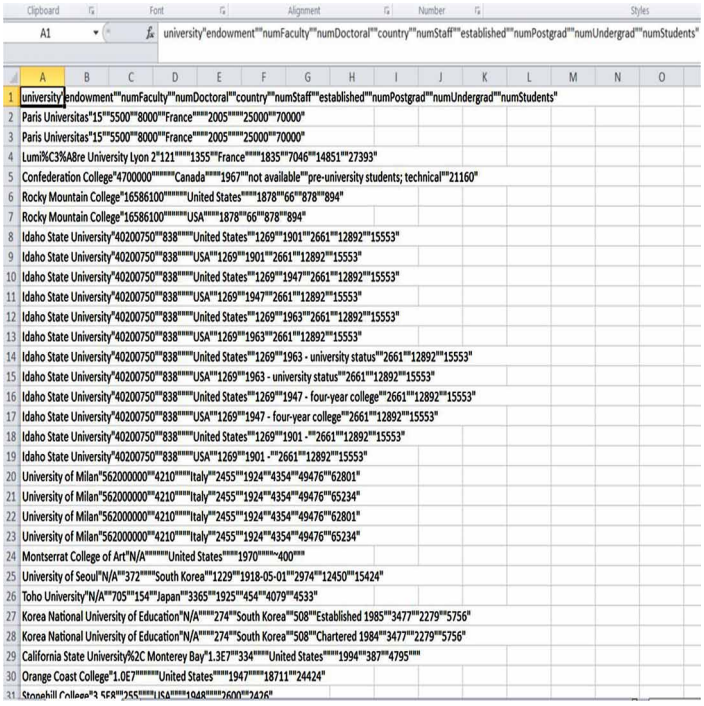
Table 7. Properties of data sets

File Name	No. of Records	No. of Fields	Missing Values	Duplicate Record
University Data	75,000	10	7.89%	32.7%
ARM Data	12,762	15	27.6%	0%

Table 8. Columns of the University Data

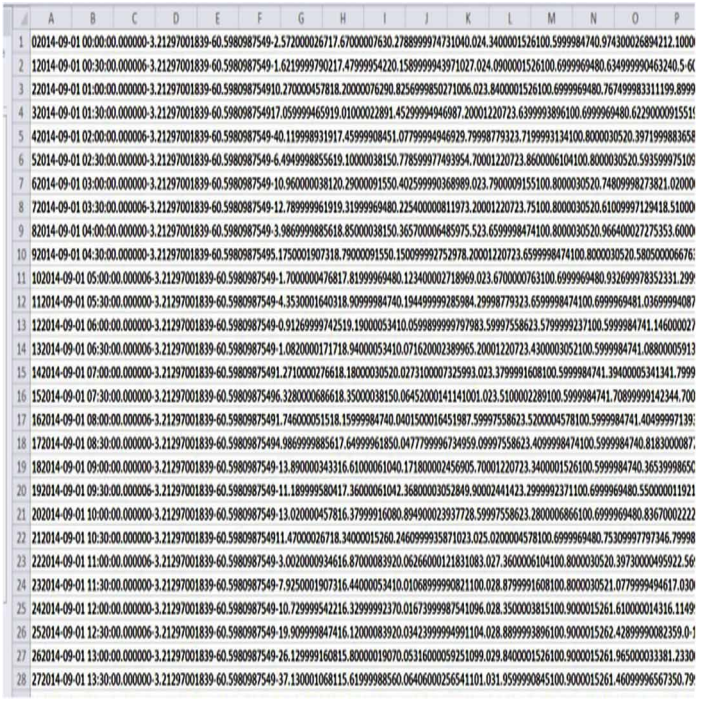
Description of Variable	Variable Name in Dataset
Name of University	University
The monetary amount of endowment the school has	Endowment
The total number of faculty employed by school	NumFaculty
Number of Doctoral	NumDoctoral
Country where the school exits	Country
The total number of staff members in the school	NumStaff
The year the school was established	Established
Number of Postgraduate students	NumPostgrad
Number of Undergraduate students	NumUndergrad
Total number of all students enrolled	NumStudents

Figure 1. Screenshot of UniversityData.csv opened with Excel



	A
1	university"endowment""numFaculty""numDoctoral""country""numStaff""established""numPostgrad""numUndergrad""numStudents"
2	Paris Universitas"15""5500""8000""France""""2005""""25000""70000"
3	Paris Universitas"15""5500""8000""France""""2005""""25000""70000"
4	Lumi%C3%A8re University Lyon 2"121""""1355""France""""1835""7046""14851""27393"
5	Confederation College"4700000""""""Canada""""1967""not available""pre-university students; technical""21160"
6	Rocky Mountain College"16586100""""""United States""""1878""66""878""894"
7	Rocky Mountain College"16586100""""""USA""""1878""66""878""894"
8	Idaho State University"40200750""838""""United States""1269""1901""2661""12892""15553"
9	Idaho State University"40200750""838""""USA""1269""1901""2661""12892""15553"
10	Idaho State University"40200750""838""""United States""1269""1947""2661""12892""15553"
11	Idaho State University"40200750""838""""USA""1269""1947""2661""12892""15553"
12	Idaho State University"40200750""838""""United States""1269""1963""2661""12892""15553"
13	Idaho State University"40200750""838""""USA""1269""1963""2661""12892""15553"
14	Idaho State University"40200750""838""""United States""1269""1963 - university status""2661""12892""15553"
15	Idaho State University"40200750""838""""USA""1269""1963 - university status""2661""12892""15553"
16	Idaho State University"40200750""838""""United States""1269""1947 - four-year college""2661""12892""15553"
17	Idaho State University"40200750""838""""USA""1269""1947 - four-year college""2661""12892""15553"
18	Idaho State University"40200750""838""""United States""1269""1901 - ""2661""12892""15553"
19	Idaho State University"40200750""838""""USA""1269""1901 -""2661""12892""15553"
20	University of Milan"562000000""4210""""Italy""2455""1924""4354""49476""62801"
21	University of Milan"562000000""4210""""Italy""2455""1924""4354""49476""65234"
22	University of Milan"562000000""4210""""Italy""2455""1924""4354""49476""62801"
23	University of Milan"562000000""4210""""Italy""2455""1924""4354""49476""65234"
24	Montserrat College of Art"N/A""""""United States""""1970""""~400"""
25	University of Seoul"N/A""372""""South Korea""1229""1918-05-01""2974""12450""15424"
26	Toho University"N/A""705""154""Japan""3365""1925""454""4079""4533"
27	Korea National University of Education"N/A""""274""South Korea""508""Established 1985""3477""2279""5756"
28	Korea National University of Education"N/A""""274""South Korea""508""Chartered 1984""3477""2279""5756"
29	California State University%2C Monterey Bay"1.3E7""334""""United States""""1994""387""4795"""
30	Orange Coast College"1.0E7""""""United States""""1947""""18711""24424"
31	Stonehill College"3,558""2,55""""USA""""1948""""2,600""2,476"

Figure 2. Screenshot of ARM data opened with Excel



	A
1	02014-09-01 00:00:00.000000-3.21297001839-60.5980987549-2.572000026717.67000007630.27889999747310 40.024.3400001526100.5999984740.97430002689421 2.1000
2	12014-09-01 00:30:00.000006-3.21297001839-60.5980987549-1.621999979021 7.47999954220.15899999439 71027.024.0900001526100.6999969480.63499999904632 40.5-6(
3	22014-09-01 01:00:00.000000-3.21297001839-60.5980987549 10.270000457818.20000076290.82569998502 71006.023.8400001526100.6999969480.76749998331 1199.8999
4	32014-09-01 01:30:00.000000-3.21297001839-60.5980987549 17.059999465919.01000022891.45299994946 87.20001220723.6399993896100.6999969480.62290000915 51
5	42014-09-01 02:00:00.000006-3.21297001839-60.5980987549-40.119998931917.45999908451.07799994946 929.79998779323.7199993134100.8000030520.3971999883658
6	52014-09-01 02:30:00.000000-3.21297001839-60.5980987549-6.4949999855619.10000038150.77859997749 3954.70001220723.8600006104100.8000030520.593599975109
7	62014-09-01 03:00:00.000000-3.21297001839-60.5980987549-10.960000038120.29000091550.40259999036 8989.023.7900091551 00.8000030520.74809998273821.020000
8	72014-09-01 03:30:00.000006-3.21297001839-60.5980987549-12.789999961919.31999969480.22540000811 973.20001220723.75100.8000030520.61009997129418.51000
9	82014-09-01 04:00:00.000000-3.21297001839-60.5980987549-3.9869999885618.85000038150.36570000485 975.523.6599998474100.8000030520.9664000272753 53.6000
10	92014-09-01 04:30:00.000000-3.21297001839-60.5980987549 5.1750001907318.79000091550.15009999275 2978.20001220723.6599998474100.8000030520.580500006767
11	102014-09-01 05:00:00.000006-3.21297001839-60.5980987549-1.700000047681 7.81999969480.12340000271 8969.023.6700000763100.6999969480.93269997835 2331.299
12	112014-09-01 05:30:00.000000-3.21297001839-60.5980987549-4.353000016403 18.90999984740.19449999285 984.29998779323.6599998474100.6999969481.03699994087
13	122014-09-01 06:00:00.000000-3.21297001839-60.5980987549-0.912699997425 19.19000053410.05998999997 97983.59997558623.5799999237100.5999984741.1460000027
14	132014-09-01 06:30:00.000006-3.21297001839-60.5980987549-1.082000017171 8.94000053410.07162000238 9965.20001220723.4300003052100.5999984741.08800005913
15	142014-09-01 07:00:00.000000-3.21297001839-60.5980987549 1.271000027661 8.18000030520.02731000073 25993.023.3799991608100.5999984741.39400005341341.7999
16	152014-09-01 07:30:00.000000-3.21297001839-60.5980987549 6.328000068661 8.35000038150.06452000141 141001.023.5100002289100.5999984741.70899999142344.700
17	162014-09-01 08:00:00.000006-3.21297001839-60.5980987549 1.746000051518.15999984740.04015000164 51987.59997558623.5200004578100.5999984741.40499997139
18	172014-09-01 08:30:00.000000-3.21297001839-60.5980987549 4.986999988561 7.64999961850.04777999967 34959.09997558623.4099998474100.5999984740.81830000877
19	182014-09-01 09:00:00.000000-3.21297001839-60.5980987549-13.890000343316.61000061040.17180000245 6905.70001220723.3400001526100.5999984740.36539998650
20	192014-09-01 09:30:00.000006-3.21297001839-60.5980987549-11.189999580417.36000061042.36800003052 849.90002441423.2999992371100.6999969480.55000001192 1
21	202014-09-01 10:00:00.000000-3.21297001839-60.5980987549-13.020000457816.37999916080.89490002393 7728.59997558623.2800006866100.6999969480.83670002222
22	212014-09-01 10:30:00.000000-3.21297001839-60.5980987549 11.470000267 18.34000015260.24609999353 871023.025.0200004578100.6999969480.75309997797346.79998
23	222014-09-01 11:00:00.000006-3.21297001839-60.5980987549-3.002000093461 6.87000083920.06266000121 831083.027.3600006104100.8000030520.39730000495922.56(
24	232014-09-01 11:30:00.000000-3.21297001839-60.5980987549-7.925000190731 6.44000053410.01068999990 821100.028.8799991608100.8000030521.07799994946 17.030
25	242014-09-01 12:00:00.000000-3.21297001839-60.5980987549-10.729999542216.32999992370.01673999987 541096.028.3500003815100.9000015261.61000001431 6.1149
26	252014-09-01 12:30:00.000006-3.21297001839-60.5980987549-19.909999847416.12000083920.03423999994 991104.028.8899993896100.9000015262.42899990082359.0-1
27	262014-09-01 13:00:00.000000-3.21297001839-60.5980987549-26.129999160815.80000019070.05316000059 251099.029.8400001526100.9000015261.96500003338 1.2330
28	272014-09-01 13:30:00.000000-3.21297001839-60.5980987549-37.130001068115.61999988560.06406000256 541101.031.9599990845100.9000015261.46099996567350.79

- Inconsistent date values. The University data contains different date formats while in the ARM data, the column for date has both date and time, which have to be separated.
- Inconsistency in abbreviations and terms used. e.g., to indicate United States of America (USA) as country, some records use terms like US, USA and United States. These might be considered as different countries without data cleaning.
- Mixture of numerical and text values.
- Missing information: The University data set contains several NA values, which is not unusual for any form of dataset but might be problematic when carrying out data analysis on dataset. While the ARM data has many missing records.
- Values in the University dataset are separated by inconsistent number of double quotes. While that in the ARM data are separated by a space which doesn't meet the comma requirement of CSV.
- Duplicate records: The dataset is supposed to have only one entry for each university instance. However, as seen in the data, some universities have several entries with all data sometimes being the same, and sometimes having variations e.g., Lamar University has 33 entries but there are variations in the values of the last variable with some showing 13773, 14388 and 14522.
- Outlier records: The ARM dataset has some values in some columns far beyond the normal. That can be problematic while carrying out analysis.
- Missing records in a sequence: The ARM dataset was collected over an interval of half an hour. There are many missing records for certain times.

The purpose of this study is to use several data cleaning tools on the same data set to compare these data cleaning tools. Through this study, it is anticipated that we will gain better insight on how these different tools work, the strength and weaknesses of each tool in data cleaning techniques as well as coming up with valuable suggestions and discussions about the future of data cleaning techniques and tools.

Tool Used

For this study, we used four different data cleaning tools namely OpenRefine, R, Python and Data Wrangler. These tools are the most popular tools used for data cleaning in the real world. OpenRefine, R and Python are open source, which makes them easily accessible for use. Data Wrangler is a commercial tool but has a community version which does a good job of data cleaning. These tools used are described below:

- **OpenRefine**: OpenRefine (Verborgh & De Wilde, 2013) is a web-based, stand-alone, open source application for data cleanup and transformation to other formats. It operates on rows of the data that have cells under columns, which is very similar to relational tables. This tool cleans, reshapes and edits batch, unstructured and messy data. It was formerly known as Google Refine and was also called Freebase Gridworks before that. Operations in OpenRefine include faceting (allowing users to narrow down results through several different dimensions), clustering, and reconciling, which all help in the data cleaning process. It also analyzes the data through filtering, faceting and converting the data into more structured form.

OpenRefine is a standalone application that has a web interface. It is not hosted on the web but can be downloaded and runs on the local machine. In other words, it is a desktop application that opens in a browser as a local webserver.

Transformation expressions can be written in General Refine Expression Language (GREL), Jython (i.e. Python) and Clojure. Since it is an open source project, its code can be reused in other projects.

OpenRefine carries our cleaning tasks through filtering and faceting, and then converts the data into a more structured format.

- **Data Wrangler:** Data Wrangler (Kandel et al., 2011) is a Stanford University project that helps analysts clean and prepare diverse, messy data quickly and accurately. It is an interactive tool for data cleaning. Data Wrangler can work with data in two ways. Users can simply paste the data into its web interface or can use the web interface to export the operations as python code and process arbitrary amounts of data. The web interface is using JavaScript and therefore has some performance issues and only supports up to 1000 rows, but users can use it to configure Data Wrangler on a subset of the data and then apply the configuration on the whole data set. The most recent version of this tool is called Trifacta Wrangler.

For the experiment we imported our data into Data Wrangler and the application began to automatically organize and structure our data set. This tool contains strong machine learning algorithms that help suggest common cleaning to be done and common transformation and aggregations. Data Wrangler allows a mixture of numerical and text values.

- **Python**: Python is another tool that can be used for data cleaning. It has several modules that can be used to carry out cleaning. One powerful module in Python that is used for data cleaning is PANDAS (Python data analysis toolkit). This module is basically for data analysis, which data cleaning is part of. Another module in Python that can be useful when carrying out cleaning is the Numpy module. This module is used for scientific computing with Python. It has a powerful N-dimensional array object that is useful for large datasets.
- **R**: R is a programming language used for statistical computation (John et el., 2016). It has been widely used for data analysis. R has a set of tools that are designed to clean data effectively and comprehensively. The R environment has the capacity to read data in several formats and process these files.

In the cleaning of data using R, four simple steps can be taken which R provides great resource for:

1. Reading data: R provides adequate reading resource from practically any format into data frame.
2. Exploratory Analysis: After reading the data, users often conduct an initial exploration of the data frame.
3. Exploratory Analysis in Visual form: During cleaning it is useful to visualize data at each stage. R provides adequate visualization tools. Three powerful visualization that can be useful during data cleaning are: Boxplot, Histogram and Scatter plot.

The Data Cleaning Tasks

In the experiment the following data cleaning tasks were conducted.

- Dealing with typographical errors or multiple representations:
  - Cleaning up inconsistent spelling of terms (i.e. “USA”, “U.S. A”, “U.S.”, etc.).
  - Converting values that are text descriptions of numeric values (i.e. $123 million) to actual numeric values (i.e. 123000000) which are usable for analysis.
  - Extracting and cleaning values for dates.
- Identifying which rows of a specific column contain a search term.
- Removing duplicate data.
- Separating date and time.
- Handling outliers.

- Handling missing records in a sequence. Here after using the tool to discover the missing records in a sequence, the user decides how to replace missing values, either by imputation, inference from other records or other method decided by the user.
- Exporting cleaned data to several formats.
- Handling missing fields, duplicate records, inconsistent formats.
- Batch editing of rows and column.

Two users with advanced programming skills finished these data cleaning tasks using the four tools. For each data cleaning task, users applied a data cleaning tool to fix the data quality issues listed in the task. They manually checked our results and repeated the cleaning task until we could not find more related quality issues in the data. The order of applying each tool for each task is randomized to avoid bias introduced by the order.

When compare the four tools, we focused on the following criteria: key features, platform, scalability, skill level needed, time of completion and ease of implementation.

RESULTS

For each tool we describe its key features, platform, skill level needed, time of completion, ease of implementation, advantages and disadvantages, accuracy.

Key Features

OpenRefine: It has the following key features:

- Importing data from various data sources and support the following format: CSV, TSV, .xls, .xlsx, JSON, XML, RDF as XML and google document. Figure 9 shows a screen shot of importing the university data using OpenRefine.
- Facets and filters: OpenRefine allow users to use facets and filters to filter data into subsets for easy usage. This can be done for numbers, text and dates columns. For example, for the University data, if a user facets data on the gender column we will get 2 in female and 1 in male. If the user selects female, then it will show the two rows with female.
- Support for expressions that can be used to create new data from existing data or transform existing data.
- Reconciliation: reconciliation matches text names or value in the columns to database identifiers in various database ID spaces. It helps resolve inconsistent spelling issues. For example, US, USA and United States can be matched to United States of America. Reconciliation can be done by calling Web Services or database API.
- Exporting Data: data can be exported into Tab separated values (TSV), Comma separated values (CSV), Excel and HTML Table.
- Undo/redo: Undo gives user the flexibility to rectify mistakes. Redo enables the user to repeat a step.

Data Wrangler: It has the following key features:

- Data Wrangler supports the following six user interactions while using the tool for cleaning.
  - Select columns
  - Select rows
  - Select text within a cell
  - Edit data within the table
  - Click bars in data quality meter
  - Assign data types, column names and semantic roles.

- Data Wrangler has a suggestion engine that suggests next data cleaning steps.
- Data Wrangler supports automated script generation.
- Data Wrangler allows user to have step by step interaction with data.
- Data Wrangler supports CSV, JSON AND TDE data formats.

Python: It has the following key features:

- Features to visualize and explore data.
- Writing customizable code for specific data cleaning tasks.
- Easy integration with other tools or product. Python can call programs written in other languages. Python code can be called in other languages as well.

R: R has the following key features.:

- R has many functions that can be used for data cleaning.
- R has good visualization libraries.
- Writing customizable code for specific data cleaning tasks.

Platform and Needed Skill Level

OpenRefine is a web-based application therefore it is platform independent. It can run on Windows, Linux and Mac. It requires basic to intermediate skill level.

Data Wrangler runs on Windows and Mac. It requires basic skill level.

Both Python and R run on all platforms, including Linux, Windows and Mac. They both require advanced skill level, because users have to know how to program.

Time of Completion

Figure 3 depicts the average completion time of cleaning using all four tools on University data and ARM data. The users have high skill level and are familiar with both R and Python.

Using Data Wrangler has the fastest completion time followed by using OpenRefine. This is expected, because both tools are highly interactive. Data Wrangler also can suggest data cleaning steps, so it leads to even faster completion time. Using R and Python took much longer time because both require customized programming. Using R took shorter time than using Python, because R has a lot of data analysis functions that are suitable for data cleaning. The relative order of different tools is also the same for both data sets.

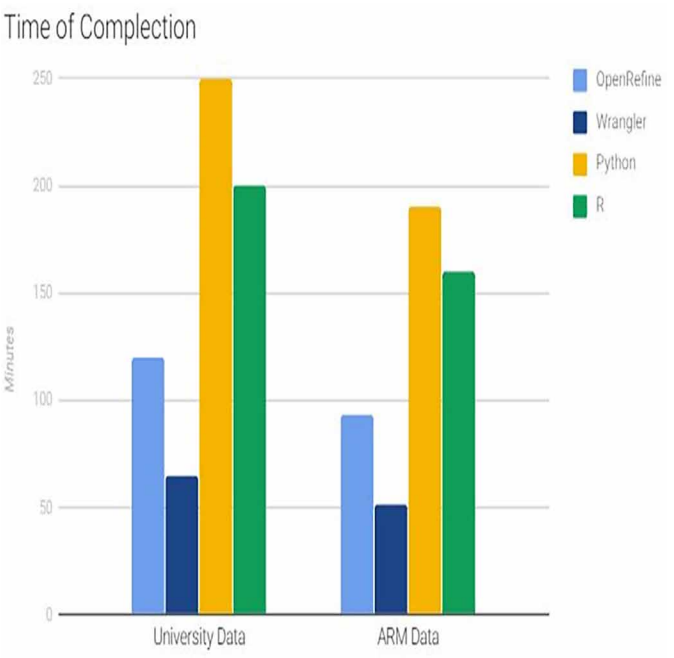
Ease of Implementation

There is no standard sequence of steps in cleaning data. Sometimes it depends on the specific issues contained in the data, while other times it depends on the user's approach. Due to this fact, we were not able to do a quantitative analysis. However, we gathered feedback from users of these tools. Some users expressed how they felt using OpenRefine and Data Wrangler for data cleaning based on the interactive user interface. Others discussed how they could use Python and R in customized ways. Based on the feedback we got and on our usage of these tools for our experiment, we assigned scale 1-3 on the ease of implementation of these tools.

We classified the ease of implementation/usage of these tools into a scale of three (3), with 3 as the easiest to use.

1. **Scale 1**: High human dependence, low interactivity, little automation and requiring advanced technical skill. This scale means that the user must know what he or she is doing, and the tool gives no suggestions or hints. The effectiveness of the tool depends on the user's knowledge

Figure 3. Time of completion in minuets using four data cleaning tools for University Data and ARM Data cleaning



and skills. In addition, the user needs to have advanced technical skills to be able to successfully complete tasks with the tool.

2. **Scale 2**: High human dependence, high interactivity, some automation and requiring basic to intermediate technical skill. Here the user is expected to know what exactly in the data he wants to clean but the tool interactively helps the user carry out the task. Basic technical skills are needed for basic cleaning, but intermediate technical skills may be needed for complex task.
3. **Scale 3**: Tools in this category are highly interactive, has little to no human dependence, and suggests cleaning steps to the user and a user with no experience nor technical skills can use this tool to achieve the data cleaning tasks.

Figure 4 shows the ease of implementation scale for each tool. OpenRefine has a scale of 2 because this tool is highly interactive and only requires basic to intermediate skills. However, users still need to specify all steps in the data cleaning process, so it has high human dependence and some automation.

Data Wrangler has a scale of 3 because it is highly interactive and only requires basic skills. In addition, it suggests data cleaning steps to users, so it has low human dependence and is highly automated.

Python and R both have a scale of 1 because they have high human dependence, low interactivity, little automation and require advanced technical skill.

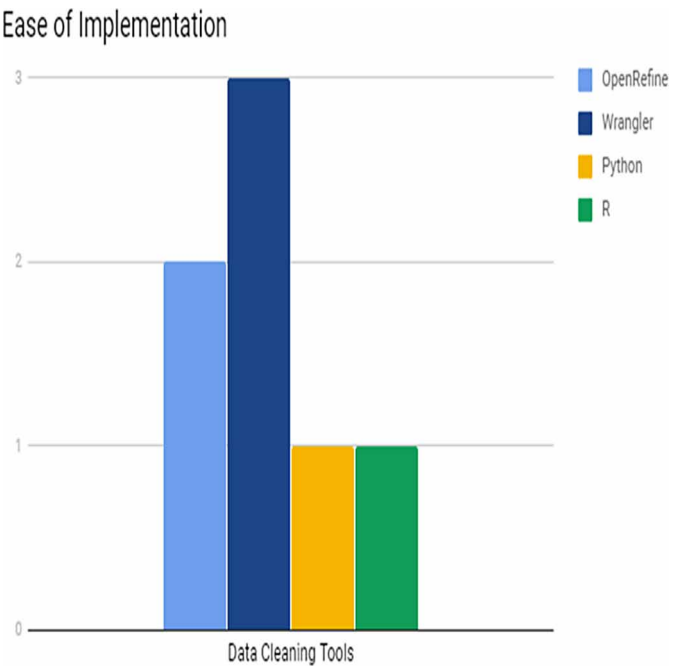
Other aspects: We looked at several other aspects, including possibility to be embed in other tools/programs, user interface, mass edits (editing multiple cells at the same time), approach, compatibility with Big Data.

Both OpenRefine and Data Wrangler are stand-alone and cannot be embedded. R and Python can be embedded in other programs.

Both OpenRefine and Data Wrangler have graphic user interface. R and Python do not.

All of them support mass editing, but R and Python require some coding to do that.

Figure 4. Ease of implementation of the tools



In terms of data cleaning approach, OpenRefine supports simple tasks as a simple click, but for more complex tasks, users need to use expression language. For Data Wrangler, users only need to click, and the system also suggests data cleaning steps. For R and Python users have to manually write scripts.

OpenRefine can only support cleaning 5000 records so it does not directly support cleaning big data. The other three tools can handle big data.

Advantages and Disadvantages

OpenRefine has the following advantages:

- Since this tool is a desktop application without the need to connect to internet, the data set is relatively safe and is harder to tamper with.
- Users can use its facet feature to filter the data into subsets.
- It has powerful features to transform data.
- It provides simple data summarization platform.

OpenRefine has the following disadvantages:

- Google removed support for this tool, and some of their features are redundant.
- The UI is not user friendly, several features are not easy to find.
- OpenRefine is not suitable for processing large data sets due to the 5000-record limit.
- It assumes that data is organized in tabular form, which is not always true.

Data Wrangler has the following advantages:

- Data Wrangler has two views. The Grid view and the Column view.
- This tool also supports data visualization and supports visualization at every step of data cleaning.
- It supports mass editing.
- It uses natural language descriptions of transformation.
- It recommends cleaning steps to uploaded data.

Overall, we found Data Wrangler the most user friendly out of the four tools.
Data Wrangler has the following disadvantages:

- It consumes lots of memory.
- Just limited features available for free version.

Python has the following advantages:

- Users can customize their solution to fit their needs.
- This tool is great as it is easy to fuse into other application.

Python has the following disadvantages:

- It requires advanced programming skills.
- The learning curve is high as it requires user to learn how to use many modules in Python.
- It's not time effective due to the high learning curve.
- This method can be complex and difficult to implement.
- Users must have previous knowledge of what steps to take in the cleaning process.

R has the following advantages:

- It is suitable when the data is mainly used for statistical analysis (e.g., sales record).
- It is very easy to visualize data at each stage of cleaning.

R has the following disadvantages:

- It is not a good option for integrating into other projects in other domains different from data science domains. Other projects might make use of other programming languages that R doesn't integrate well with.
- It's not time effective due to the high learning curve.
- This method can be complex and difficult to implement.
- Users must have previous knowledge of what steps to take in the cleaning process.

Accuracy

We were not able to quantify the accuracy of these tools, because users have to go through multiple iterations for each tool and once some issues are fixed in one iteration, the tool may find more issues that will be fixed in the next iteration. In the experiments, we observed OpenRefine and Data Wrangler to have high accuracy when detecting specific data quality issues (e.g., missing values). But some manual work is needed to fix the found issues (e.g., you can decide to remove missing values or assign some values).

For R and Python, the accuracy all depends on the user's skill level, whether the user can write good programs to detect those issues and solve them.

At the end, most data quality issues are addressed for each tool.

Summary of Comparison

Table 9 summarize the comparison of these four tools. Following comparison criteria used by Porwal & Vora (Porwal & Vora, 2013) and Karrar and Ali (Karrar & Ali, 2016), we came up with the following metrics: import format, performance time, skill level, platform, ease of implementation, key features, output format, skill level, platform, accuracy, possibility to be embed in other tools/programs, user interface, mass edit, approach, compatible with big data and their disadvantages.

Table 9. Comparison of OpenRefine, Data Wrangler, Python and R

Criteria	OpenRefine	Wrangler	Python	R
Import format	CSV, TSV, Excel (XLS/XLSX), JSON, XML, RDF	Excel (XLS/XLSX), CSV, TEXT	All	All
Performance time	Depends on data size and format	Depends on user choice and data size	Depends on user programming skills and level of artifact in data	Depends on user programming skills and level of artifact in data.
Key features	Facets and filters, Support for expression language, Reconciliation	User interactions, suggestion engine, automated script generation,	Customizable by user, integrate with other tools, great visualization library	Customizable by user, great visualization library
Skill level	Basic to Intermediate	Basic	Advanced	Advanced
Platform	All platform	Windows, Mac	All platform	All platform
Accuracy	High accuracy when detecting specific data quality issue	High accuracy when detecting specific data quality issue	Depends on the user's skill level	Depends on the user's skill level
Platform	All platform	Windows, Mac	All platform	All platform
Ease of implantation	2	3	1	1
Output format	TSV, CSV, Excel and HTML Table	CSV, JSON, TDE	User may customize to any format	User may customize to any format
Possibility to embedded	No, Standalone but code is available	No, Standalone	Yes	Yes
Graphic User Interface	Yes	Yes	No	No
Edit Multiple Values	Support mass edit	Support mass edit and its easy	Support but require complicated coding	Support but require coding
Approach	Simple task carried out with a click, but complex task requires expression language	Simple click and also suggest cleaning features for user	Need to write scripts	Need to write scripts
Compatible with Big Data	No (suitable for only 5000 records)	Yes	Yes	Yes
Draw backs	Google stopped support, advanced features require technical skills	Memory consumption is high, cost implication	Require good knowledge of programming	Require knowledge of programming and statistics

CONCLUSION

The problem of 'dirty' data costs institutions large amounts of money every year. More than 80% of time and resources are spent preparing and cleaning data. This paper conducted a comparison study of four commonly used tools for data cleaning. The results show that OpenRefine, which is an open source tool developed by Google, is a useful tool and has several merits such as the feature of running locally which makes user data more secure, and the feature with a graphical interface and the mass edit feature. But OpenRefine needs experience and expertise to be able to use its advanced features. OpenRefine also works better for small data sets.

Data Wrangler has the advantage of being a standalone tool. It is very efficient for big data and has a unique visualization feature at each step and gives the user an opportunity to preview changes made graphically before committing the change. It can also recommend data cleaning steps. Overall it is the easiest to use. However, the free version has limited functionalities.

Python and R have the advantage for the user to customize the data any way s/he wants, and they can be embedded into other tools. Both Python and R have same features in cleaning, but Python has lots of modules to support different aspect of cleaning and the ability to use this data for other analysis. Python and R however, require great programming skills, which may not be present. Python and R also take lots of time to carry out cleaning as each step along the way must be implemented manually.

In conclusion, Data Wrangler will be a good start for novice user, as many data analyst will prefer not to spend too much time cleaning data, as they must work on the functionality or usage of these data. It will be good for users that don't mind paying for cleaning tool. For users seeking open source tool, OpenRefine is a good option. For data engineers that have time and adequate skills, Python or R will be a good option.

One possible future work is to take each tool and look at how it can help us in the integration of data from diverse sources.

REFERENCES

Batini, C., Lenzerini, M., & Navathe, S. B. (1986). A comparative analysis of methodologies for database schema integration. *ACM Computing Surveys*, *18*(4), 323–364. doi:10.1145/27633.27634

Castanedo, F. (2013). A review of data fusion techniques. *The Scientific World Journal*. PMID:24288502

Dasu, T., & Johnson, T. (2003). *Exploratory data mining and data cleaning* (Vol. 479). John Wiley & Sons. doi:10.1002/0471448354

Galhardas, H., Florescu, D., Shasha, D., & Simon, E. (2000). *AJAX: an extensible data cleaning tool*.

Haghighat, M., Abdel-Mottaleb, M., & Alhalabi, W. (2016). Discriminant Correlation Analysis: Real-Time Feature Level Fusion for Multimodal Biometric Recognition. *IEEE Transactions on Information Forensics and Security*, *11*(9), 1984–1996. doi:10.1109/TIFS.2016.2569061

John, F., & Allison, L. (2016). R and the Journal of Statistical Software. *Journal of Statistical Software*, *73*(2).

Kandel, S., Paepcke, A., Hellerstein, J., & Heer, J. (2011). Wrangler: Interactive visual specification of data transformation scripts. *Paper presented at the Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*.

Karrar, A. E., & Ali, M. M. (2016). Comparative Analysis of Data Cleaning Tools Using SQL Server and Winpure Tool. *International Journal of Computer Applications in Technology*, *3*(7), 371–377.

Kumar, S., & Nadeem, M. (2008). Extraction, Transformation, Loading (ETL) and Data Cleaning Problems. *Journal of Independent Studies and Research on Computing*, *6*(1).

Lee, M. L., Lu, H., Ling, T. W., & Ko, Y. T. (1999). Cleansing data for mining and warehousing. *Paper presented at the 10th International Conference on Database and Expert Systems Applications*.

Martinez-Mosquera, D., Luján-Mora, S., López, G., & Santos, L. (2017). Data Cleaning Technique for Security Logs Based on Fellegi-Sunter Theory. *Paper presented at the SIGSAND-EuroSymposium*, Gdansk, Poland.

Müller, H., & Freytag, J.-C. (2005). *Problems, methods, and challenges in comprehensive data cleansing*.

Parent, C., & Spaccapietra, S. (1998). Issues and approaches of database integration. *Communications of the ACM*, *41*(5es), 166–178. doi:10.1145/276404.276408

Patel, S. (2012). Requirement to cleanse DATA in ETL process and Why is data cleansing in Business Application? *International Journal of Engineering Research and Applications*, *2*(3).

Porwal, S., & Vora, D. (2013). A Comparative Analysis of Data Cleaning Approaches to Dirty Data. *International Journal of Computers and Applications*, *62*(17).

Rahm, E., & Do, H. H. (2000). Data cleaning: Problems and current approaches. *IEEE Data Eng. Bull.*, *23*(4), 3–13.

Vassiliadis, P., Simitsis, A., & Skiadopoulos, S. (2002, November). Conceptual modeling for ETL processes. In *Proceedings of the 5th ACM international workshop on Data Warehousing and OLAP* (pp. 14-21). ACM.

Verborgh, R., & De Wilde, M. (2013). *Using OpenRefine*. Packt Publishing Ltd.

Samson Oni is a PhD student of Information Systems in the University of Maryland Baltimore County (UMBC). He obtained his master's degree in computer science University of Maryland, Baltimore County. He worked as a Research Assistant at the Imaging Research Center UMBC. His previous work includes technical intern for Joint Centre for Earth Systems (NASA-JCET) - UMBC and Full-stack developer for Department of education UMBC. His research focus is in cyber security and Data science and have carried out several projects in these domains. Currently, he is a research assistant at the Information Systems UMBC where he is working on semantic web, blockchain and cybersecurity-related projects. More information can be found at http://www.samdwise.com

Zhiyuan Chen is an Associate Professor in Department of Information Systems at University of Maryland Baltimore County. He received a PhD degree in Computer Science from Cornell University in August 2002. He has more than 10 years of extensive research experience in data privacy, privacy preserving data mining, database management, data science, and cyber security. His main research focus is in algorithms for preserving privacy of data and at the same time allows accurate analysis of the data. He has published over 40 papers in peer reviewed journals and publications and over 20 of them are in the area of privacy and security. More information can be found at https://userpages.umbc.edu/~zhchen/

Susan Hoban worked with NASA for over two decades, first as a scientist studying comets and the interstellar medium, then as a STEM Educator. Dr. Hoban develops curriculum for professional development of educators for classroom use and informal education venues. Dr. Hoban specializes in integrating hands-on activities with data collection and analysis to develop the habits-of-mind of STEM. Curriculum modules include, but are not limited to rocketry, environmental education, astronomy & astrobiology, computer modeling, STEM music, and robotics for learners of all ages. Dr. Hoban is currently also working on using analytics for cyber security.

Onimi Jademi is a PhD candidate in the Department of Information Systems at the University of Maryland, Baltimore County (UMBC). Her research focuses on natural language processing and machine learning, and its applications especially in the healthcare domain. She has experience with high quality qualitative and quantitative research methods.