A Qualitative Assessment of Machine Learning Support for Detecting Data Completeness and Accuracy Issues to Improve Data Analytics in Big Data for the Healthcare Industry

Suraj Juddoo
School of Digital Technologies
Middlesex University Mauritius
Cascavelle, Mauritius.
s.juddoo@mdx.ac.mu

Carlisle George
School of Science and Technology,
Middlesex University,
London, UK.
c.george@mdx.ac.uk

***Abstract*—Tackling Data Quality issues as part of Big Data can be challenging. For data cleansing activities, manual methods are not efficient due to the potentially very large amount of data.. This paper aims to qualitatively assess the possibilities for using machine learning in the process of detecting data incompleteness and inaccuracy, since these two data quality dimensions were found to be the most significant by a previous research study conducted by the authors. A review of existing literature concludes that there is no unique machine learning algorithm most suitable to deal with *both* incompleteness and inaccuracy of data. Various algorithms are selected from existing studies and applied against a representative big (healthcare) dataset. Following experiments, it was also discovered that the implementation of machine learning algorithms in this context encounters several challenges for Big Data quality activities. These challenges are related to the amount of data particualar machine learning algorithms can scale to and also to certain data type restrictions imposed by some machine learning algorithms. The study concludes that 1) data imputation works better with linear regression models, 2) clustering models are more efficient to detect outliers but fully automated systems may not be realistic in this context. Therefore, a certain level of human judgement is still needed.**

Keywords: Big Data, Data Quality, Data Inaccuracy, Data incompleteness, Machine Learning.

I. INTRODUCTION

Big Data analytics may be ineffective if the raw data is of poor quality (for example. inaccurate, incomplete, inconsistent and unreliable). Hence there is the need to perform pre-processing, standardization and cleaning activities to improve the standard of data quality. Previous literature has described the use of data mining and statistical based methods to improve data quality, for example, improving *accuracy* of data by trying to predict and fill missing values in datasets [1]. Also, there has not been much previous research focusing on the use of machine learning algorithms to improve data quality in Big Data.

This current research, however, argues that machine learning (ML) can be a very efficient and effective tool to improve data quality specifically in the Big Data context by detecting bad quality data, in terms of incompleteness and inaccuracy.

For the data 'completeness' Data Quality Dimension (DQD), detection of missing values in a dataset is very simple to carry out using data science tools such as RapidMiner Studio. Conversely, ML algorithms can be applied to solve 'completeness' issues through imputation techniques [3],[4]. As for the 'accuracy' DQD, some ML algorithms have been proposed to deal with different accuracy issues such as noise and outlier detection [15],[17]. Since using ML algorithms to detect completeness and accuracy issues is a relatively new research area, there is need for a more systematic and complete review, coupled with experiments in the specific area of Big Data for healthcare. Therefore, this current paper aims to focus on investigating the degree of support given by ML algorithms for ***detecting*** data completeness and accuracy issues within a Big Data context as part of the healthcare industry.

This paper first discusses the current state of knowledge pertaining to the use of machine learning to tackle data completeness and accuracy issues in general. Next, this state of knowledge is validated by applying relevant machine learning algorithms (as discussed within the literature below) on a Big Data example from the healthcare industry. Finally, clear and accepted evaluation measures and metrics are applied to identify the best machine learning algorithms for detecting data completeness and data accuracy issues.

3rd International Conference on Emerging Trends in Electrical, Electronic and Communications Engineering (ELECOM). University of Mauritius, 25-27 November 2020. ISBN 978-1-7281-5706-1

II. LITERATURE REVIEW

A. Completeness DQD

The *completeness* DQD is expressed in different ways, such as missing values, absent values and sparseness of values. One of the main goals of data pre-processing is catering for missing values to provide high quality data and ultimately maximize value from the analytics. The use of the **Bayesian isotonic regression algorithm** was proposed in a past research study for medical data cleaning focusing on blood tests data [4]. The hypothesis was that missing values can be filled with predicted values based on historical data. From that research study, some aspects remain unclear such as the correctness of the imputed values used to replace the missing values, and the percentage of missing values being imputed with this algorithm. One of the generally reported issue of data quality correction techniques is the possibility of new errors being introduced while correcting detected errors [5]. Thus, use of ML for imputing missing values needs to be evaluated in terms of the correctness of imputations performed.

Most services offered by Intelligent Transportation Systems (ITS) depend on accurate and complete data [6]. For ITS, missing values originate mostly from issues with sensing equipment and transmission network, with up-to 56% of reported missing data in an ITS used in Melbourne (Australia). Different missing value imputations (MV) methods have been proposed throughout past research literature, and they cite principally ML models such as K-Nearest Neighbours (KNN), Singularity Value Decomposition (SVD), Probabilistic Principal Component Analysis (PPCA) and Low Rank Matrix Completion (LRMC). A novel self-representation based matrix completion approach for missing data recovery by incorporating l_p-norm regularised sparse self-representation (SRS-l_p) was proposed by Chen et al [6]. Traffic flow data is spatially and temporally correlated with one another, which lends itself favourably towards solutions such as a self-representation based matrix. ***However, this spatial and temporal correlation might not exist for the healthcare data context as part of this current research.*** In Chen's study[6],many ML algorithms such as KNN, PPCA and LRMC were compared with SRS-l_p. Missing values were artificially introduced in different conditions and a root mean squared error (RMSE) was used to measure missing value recovery performance [6]. The results showed that the Local Least Squares (LLS) algorithm was more effective with low missing ratios, but with higher missing ratios, the self-representation matrix method produce better results.

In past literature, the imputation of missing numerical values are reported to be undertaken by statistical processes and therefore does not absolutely require the use of machine learning models. For example, in a study involving network based data, different statistical methods to impute missing values were proposed [7], such as:

a) Random imputation from last 30 measurements

b) Random imputation from last 30 measurements of second differences

c) Impute using previous non-missing values

d) Impute average of last 3 non-missing values

These methods are suitable in datasets consisting primarily of numerical missing values, which could work well in some settings of healthcare data such as laboratory data, but will not be effective with health data containing text data.

Yu et al [3] proposed a modification of the K-NN imputation algorithm, known as Cluster-Based Best Match Scanning (CBMS) in terms of improved computational complexity and improved space/memory usage with comparable level of accuracy to K-NN . Simulation was carried upon a large smart meter reading dataset and imputation testing accuracy was measured using the mean absolute deviation method. Over and above the computational complexity and memory usage improvements, CBMS proved to be a model which can work with parallel computing.

The multivariate Regressor algorithm has also been discussed as a potential method to deal with missing values [8]. This was one of the earliest examples of a single imputation model which is trained on a sample of training data without any missing values. . Lin et al [9] concluded that random hot deck imputations, coupled with oversampling and bootstrap methods were more effective in clinical datasets with different types of missing values deficiencies . However, the experiments carried out made use of small datasets on hundreds of tuples only. Hence, it is quite unclear what would be the performance of hot deck imputations for Big Data scenarios. Zhang et al [10] proposed the Clustering Based Random Imputation (CRI) method to overcome lack of efficiency of other imputation techniques such as Nearest Neighbour imputation. The original dataset was divided into two: one without missing values and one containing all instances of missing values. Those datasets were further subdivided into clusters using K-Means to group instances into clusters. Each instance with missing values was matched to clusters containing complete values similar to it using Euclidean distance calculation. A kernel based function was used to impute the missing values in one instance using the comparable information coming from instances and attributes of the matched cluster without missing values The reports of the techniques discussed above lacked sufficient details of their implementations and evaluations, and therefore warrant further investigations.

The use of clustering techniques combined with feature selection was promoted to reduce computation time of both single and multiple imputation methods, while also improving accuracy for classification [11]. The differences between this current research study and the one undertaken by [11] are: (1) the goals of the latter was to

improve completeness DQD for further classification work upon the datasets, whereas this current research study aims at improving the level of incompleteness of a dataset, irrespective of its subsequent use, (2) the datasets used by [11] were very small datasets, whereas this current research study situates data quality in a Big Data context, and aims to carry out experiments upon datasets demonstrating the volume characteristic of Big Data. However, despite the differences denoted, reduction of computation time was considered to be a very worthwhile feature in a Big Data context, and therefore, the use of clustering machine learning algorithms was investigated as part of being adequate techniques to improve data cleansing methods.

B. Accuracy DQD

The accuracy DQD is exemplified as Data Quality (DQ) errors at the instance level of a database in some cases. Examples of these errors are missing data, incorrect data, misspellings, ambiguous data, outdated temporal data, "misfielded" values and incorrect references [12]. Due to this variety of 'accuracy' errors, there is a need to investigate which ML algorithms could most efficiently detect and classify the types of errors mentioned above. Some of these DQ errors are semantic in nature, such as "misfielded" data, which refer to data values that are inserted in improper data attributes. An example of this would be a first name value inserted in a Surname data attribute. Thus, these semantic errors are difficult to detect automatically and might require some level of human intervention to validate proposed error detections.

Probing deeper into the accuracy DQD reveals that there is a cluster of dimensions of accuracy [12]. *Structural accuracy* refers to the general idea of the closeness of a value to a real-life phenomenon. *Syntactic accuracy,* which is a sub-type of structural accuracy, refers to the closeness of a data value to elements in a corresponding domain. For example, even if the true Author of a book might be Mr John, but it is recorded as Mr Jack, it might not be syntactically inaccurate if the value 'Jack' forms part of acceptable domain of names. Finally, *semantic accuracy,* another sub-type of structural accuracy, refers to the closeness of a data value to its true value. An example of this would be whether Mr John is the real author of book 'X', and thus refers mostly to incorrect data, which matches closely the correctness DQD [13]. This shows the differences of terms used to denote DQ issues within existing literature. The rapidity with which changes in real-life phenomenon is updated upon data values is known as *temporal accuracy.* Hence, detection of those different types of accuracies need different techniques. This current research investigates whether a single unique ML algorithm could efficiently address all of those types of accuracy errors or whether different ML algorithms are required, hence making accuracy issues detection a computationally expensive task.

Rahman et al [5] criticised the use of the 'class attributes' technique as the foundation method for determining noisy values either as part of attributes or records in a dataset due to the fact that there are always some exceptional records that cannot be classified accurately. The same authors further assert that there is typically a low amount of noise as part of datasets. This statement contradicts ideas proposed by other authors who argue that datasets typically contain a high amount of errors and therefore require much effort by data scientists for data pre-processing activities. Rahman et al [5] proposed a technique known 'CAIRAD: A Co-appearance based Analysis for Incorrect Records and Attribute-values Detection', but it depends upon correlation of noise between attribute values; however, the authors explained that many errors are random and independent, such as typo errors, and therefore no correlations exist. Finally, 'CAIRAD' is intended to work only on numerical attributes. Hence, the above-mentioned reasons support the aim of the current research work to investigate other techniques that would improve data accuracy rates for Big Data in the health industry.

Yuan et al [14] undertook a comparative analysis of three machine learning algorithms namely Support Vector Machines (SVM), naïve bayes and Gradient Boosting Decision Tree(GBDT) to detect data faults in wireless sensor networks . There exists many reasons why sensor nodes might produce faulty data, and those include harsh environmental conditions and poor calibration of sensors among others. The faults were subdivided into three subtypes, namely noise, fixed and short term faults; these sub-types were artificially introduced into the experimental dataset used for that research. As part of the evaluation of the three ML algorithms, Yuan et al. [14] used true positive rate (TPR), false positive rate (FPR), detection accuracy (DA) and precision as benchmarks. They concluded that GBDT outperformed the two other ML algorithms for the three sub-types of faulty data under consideration. Yuan et al [14] did not use Big Data, however, the methodology used can be replicated for the evaluation of different ML algorithms, and created a strong case to consider GBDT as one ML algorithm to use in the current study's experiments.

The presence of 'noise' as part of datasets may result in DQ accuracy problems. The main methods used to detect and remove noise are binning, clustering and regression [15]. *Binning* involves smoothing out values in a group (bin) by substituting some 'noisy' values with the mean or median value of the bin. The validity of using the smoothing value becomes highly questionable and difficult to implement in the case of real time streaming data, as could be the case for Big Data. *Clustering* involves detecting irregular pattern in a dataset by choosing an appropriate model. The issue is that this technique is efficient for datasets containing homogenous data, which might not be the case for Big Data due to the variety characteristic. With *regression,* outlier data is smoothed out via the use of a proper smoothing algorithm (linear, multiple or logistic). A general point of concern is the high cost of

data cleaning for Big Data, and therefore prohibitive for complex classifiers [15]. Thus, apart from the accuracy and recall evaluation measures, processing time was considered as another evaluation criteria for this current research study.

Wu and Zhu [16] proposed two main methods to deal with the problem of noisy data: 1) applying data cleansing methods to eliminate data quality issues as far as possible, and 2) make data mining applications more robust so that they can tolerate the presence of noisy data. The first method presents some drawbacks, such as: (1) data cleansing algorithms deal with only certain types of errors,(2) data cleansing cannot result into perfect data, (3) data cleansing cannot always be applied to all data sources, (4) eliminating noisy data may lead to crucial data loss for further mining/analytics and (5) the data mining/analytics algorithm cannot consider the original data source context after data cleansing has been applied. However, making data mining applications more tolerant towards the presence of noisy data is based upon a very important assumption, that there is sufficient knowledge of the type of errors that are present as part of a dataset before the actual analytics is applied. This might hold true in several cases (known device errors, known information transformation errors), but lack of knowledge of provenance of data source is a very high possibility with Big Data..

Typos represent another category of data accuracy issues. Yinghao et al [17] proposed Neural Networks classifiers, coupled with knowledge bases as an efficient method to detect typos. The knowledge bases involved were general English dictionaries such as WinEdt, commonly misspelled words aggregated together by Wikipedia and domain specific lexicon regarding vehicle diagnostics.. The Neural Network was trained with a set of misspelled words and their correction candidates. This step was useful in order to select the most precise replacement whenever a typo had been detected. Experimental evaluation of the proposed methodology against Google Spell checkers and Aspell Check showed a much better performance in terms of rate of detection and more precise corrections by the proposed system[17]. In the light of the the methodology in [17], the current research determined that it is not realistic for the Big Data context, as the existence of a dictionary for health jargon is not always available..

Sporleder et al [18] proposed vertical and horizontal error correction methods as part of semi-automatic error detection tools for text data . Horizontal error correction aims at identifying and correcting errors within a database record whereas vertical error correction aims at doing the same for values inserted in incorrect columns, described as 'misfielded' errors. The methods used were data driven and language independent, which expands their range of applications. Also, even if supervised machine language algorithms were used, the authors claimed that there is no need for manual annotation since the training set would be obtained from the database itself. ***This fact is highly interesting and relevant for this current research, which could make it appropriate to use supervised learning algorithms even if no prior training set is available.*** Precision and Recall were used as evaluation measures and the results were very satisfactory. The test database used as part of the evaluation was quite voluminous and highly dimensional, which again is very similar to what this current research is also aiming for. The techniques used are association rules for horizontal error detection and TF-IDF for vertical error detections.

III. EXPERIMENT DESIGN

There are several tools available for the application of ML algorithms to classify data. Some examples are WEKA, RapidMiner Studio and Python libraries such as Scikit. After a review of different possibilities, RapidMiner was initially selected since it allows extremely fast and easy ML deployment upon different types of data sources.Furthermore, based on reviews from insitutions such as Gartner[21], RapidMiner is cited as one of the best industry tools for data science and ML solutions.

A. Dataset considered

A CSV formatted dataset was selected for carrying out the data quality detection and transformation experiments as it embodied Big Data characteristics and metadata analysis showed some DQ issues. This dataset was freely obtained and downloaded from 'www.healthit.gov'. The title of the dataset was 'EHR Products Used for Meaningful Use Attestation' and contained data about vendors, products, US health provider specific data and other general public or non-private data. An online document provided metadata about the dataset and specifyed the different attributes and the attribute descriptions. Those details were essential in order to understand data quality issues, and in the context of this research study, accuracy and completeness issues for all the 23 attributes as part of the dataset. The dataset itself consisted of 1,048,576 rows of data, combined with the 23 attributes, was a large dataset in terms of the 'volume' characteristic of Big Data.

The dataset was connected as a local data repository with RapidMiner Studio, and the statistics feature revealed completeness issues in terms of missing values. For example, an attribute named 'CCN', which was a unique identifier for health care facilities certified to participate in federal health care programs, was reported to have a staggering amount of 1009941 missing values. Other attributes, such as 'Speciality', reported less missing values of only 38633.

To deal with missing values, several features are available in terms of filling data gaps, imputing missing values and replace missing values with some existing tools such as RapidMiner Studio.The filling of the data gaps feature to deal with missing values is less relevant for this current

research as missing ID attributes values only are calculated based on the greatest common divisor of distances of consecutive IDs. The other attributes would have been filled with a null value. Likewise, the 'replace missing values' process is not considered highly applicable as missing values is replaced by a specific replacement such as a minimum, average or maximum value of a given attribute.

To tackle data completeness issues, the 'impute missing value' operator was used . Within this operator, there is a sub-process, which takes the repository as input and applies the K-NN algorithm to derive value imputations. This K-NN model replaces missing values by using Euclidean distance relative to available data to 'guess' more precisely what missing values could be. The first experiment involved the selection of all attributes to be imputed across all tuples in the dataset. However, upon execution of this process, the computation period went for 24 hours without achieving any output. The process was arbitrarily terminated and a subset of attributes was performed, filtering out the 'CCN' identity attribute and including only attributes (NPI, zip, Provider_type) which could be correlated with the 'hospital type' attribute. The latter is one attribute which displayed some units of missing values. However, even reducing the dimension to only four attributes resulted in the imputation process running without any output, before being arbitrarily terminated .

The authors suspected that the number of rows or examples involved, i.e. over 1 million, was a challenge to the computation capability of the software using a local desktop processing capability system. This suspicion was confirmed when the examples were filtered, taking only examples from range 2000 to 2100. Hence, the imputation process was applied on only 100 observations, with four attributes involved, using a kNN algorithm. However, this resulted in an error message saying implicitly that the amount of memory available was not sufficient to run this process. For all these reasons discussed above, the use of RapidMiner Studio was judged to be inadequate for working with big datasets; hence, out-of core learning was recommended as part of the ML based solutions. Out-of core learning algorithms tolerate working with data too big to fit in the RAM of a computer system, and python offers the incremental learning possibility with the 'partial-fit' API.

Python is a well-known programming language and it is used extensively within the data science community. It possesses some interesting libraries such as 'scikit.learn' and 'impyute' to help deal with data quality issues. The first use of python for the current research was an exploratory data analysis, with the aim of detecting attributes having missing values and their quantity. The result was as follows:

Table 1:Exploratory data analysis

Attribute	**# of missing values**
NPI	0
CCN	1009941
Provider_Type	0
Business_State_Territory	0
ZIP	40845
Specialty	39237
Hospital_Type	1009941
Program_Type	0
Program_Year	0
Provider_Stage_Number	0
Payment_Year	76340
Attestation_Month	0
Attestation_Year	0
MU_Definition_2014	846028
Stage_2_Scheduled_2014	23250
EHR_Certification_Number	0
EHR_Product_CHP_Id	0
Vendor_Name	0
EHR_Product_Name	0
EHR_Product_Version	0
Product_Classification	8069
Product_Setting	8069
Product_Certification_Edition_Yr	0

Hence, it was clear that there were varying amounts of missing values across different attributes.With our example, it can be inferred that there is a correlation between 'CCN' and 'Hospital_Type', and between 'Product Classification' and 'Product Setting'. Feature reduction is understaken to reduce the computational complexity of imputation missing values. Hence, 'CCN' is not considered for imputation.

The next step was data auditing. In order to know whether the experimental dataset faces inaccuracy issues, simple statistical analysis upon the datasets using counts of values per attribute, mean, standard deviations, frequency, minimum and maximum values were carried out. This revealed with better clarity the accuracy problem/s which certain attribute/s might be facing. This knowledge needed to be coupled with the general context of the dataset, in order to correctly discriminate between errors and acceptable extraordinary values.

The logic or pseudocode for predictive modelling for imputing missing values in selected features used in this paper is as follows:

Call the variable where you have missing values as y.

Split data into sets with missing values and without missing values, name the missing set X_test and the one without missing values X_train and take y (variable or feature where there is missing values) off the second set, naming it y_train.

Use one of the discussed ML algo. to predict y_pred.

Add it to X_test as your y_test column. Then combine sets together.

This research focused on detecting human or mechanically induced errors as part of the considered dataset. The result of an extreme value analysis, applicable only on numerical data, for detecting inaccuracy issues were as follows:

Table 2: Extreme value analysis results

Feature	% of Outlier detected
CCN	0
ZIP	9
Program_Year	0
Payment_Year	0
Attestation_Month	10.2
Attestation_Year	0
MU_Definition_2014	0
Stage_2_Scheduled_2014	0
Product_Certification_Edition_Yr	0

Table 2 above clearly demonstrate issues with only 'ZIP' and 'Attestation_Month' features. These outliers are not necessarily errors, hence there will be the need to have human inspection of those outliers only and then identify errors.

The following sections highlights the different experiments carried out using the algorithms identified through the literature review. For imputation experiments, the 'Payment_Year' attribute was selected, whereas for detection of outliers for text values, experiments were performed upon 'Speciality' and 'Program_Type' attributes.

1) Bayesian isotonic regression

From the details given in the research study consulted [4], it was not possible to have a complete source code and which could guarantee correct replication of the Bayesian isotonic regression implementation. There were also issues in terms of the need to have some 'historic set of values' which should facilitate training. Hence, it was decided to implement a very close alternative in terms of 'isotonic regression' algorithm. However, this algorithm was found to be inadequate for Big Data as it cannot cater for the volume of data. As regression algorithms were commonly cited in existing research studies [3-6], linear regression was implemented as it is described as quite close to isotonic regression algorithms [4]. The linear model class from sklearn library in python 2.7 was implemented with the data split as described in the pseudocode for imputation of missing values above.

2) SRSp

Following the guidelines of the original research study, it was again impossible to have a perfect replication due to missing details of the implementation. However, following the algorithm documentation, the main functions of the algorithm is based on the application of sparsity and regularization functions. In order to achieve this, an SDRegressor class of sklearn library in python 2.7 was implemented, with different parameters fine-tuned to have an emulation of SRSp.

3) Cluster-Based Best Match Scanning

This algorithm was again not properly explained as part of the original research study[3], but the logic understood from the algorithm implies the need to have regression of missing values performed upon clusters of data. Hence, k-means algorithm was applied for clustering and KNN for regression. The number of clusters, denoted by k, was set to 6 to have a better clustering of data points. Then, each cluster was split into 70% as training and 30% test set. Unfortunately, KNN algorithm cannot accommodate more than 1 million rows of values, and hence CBMS was deemed to be inadequate for Big Data.

4) Clustering combined with TF-IDF

Detecting errors in text data can be achieved using ML clustering such as *k-means*. However, as *k-means*, or any other clustering algorithm, cannot be applied on text data directly, there is the need to convert the text data into numerical data. During this conversion, each word is assigned a weight approximating its importance in a group of documents. For the current experiment, as there were no typographical or grammatical errors present as part of the attributes containing text data, errors were artificially introduced as part of two attributes. Then, the "term frequency-inverse document frequency" (TF-IDF) algorithm was applied upon each attribute to produce a weight for each word. As there were many values that were repeating within an attribute, the *tf-idf* of these values were similar.

Following the transformation of text values into a series of *tf-idf* values, *k-means* algorithm was applied with the creation of only one cluster. The least important values were easily identified. For example, when applying this method on the 'Program_Type' feature, the experimental algorithm output was: "***Cluster 0: medicare medicaid hegfgf medigfgf".***

The last two values were artificially induced errors. Thus, with the application of the *k-means* algorithm with a single cluster upon the 'tf-idf' equivalents of words in a dataset, errors will normally be outputted amongst the last

in a cluster. Afterwards, human intervention will be necessary to ascertain whether those last values are valid one or errors.

IV. FINDINGS

The following table summarizes findings made after attempting to implement ML algorithms in Big Data for the healthcare industry, as discussed as part of the literature review.

Table 3:Summary of-findings

ML Algorithm	Findings following implementation
Bayesian isotonic regression	Implementation in python exists only for isotonic regression, but works only with 1d array as parameters, hence, not functional for multiple features; hence, had to implement a close alternative in the form of a linear regression algorithm.
lp- norm regularisation (SRSp)	SDRegressor library of the linear model was implemented. This model allows application of sparsity through the 'penalty' and 'l1_ratio' parameters and parameter 'alpha' which allows regularization. Those two parameters are the foundational building blocks of the SRSp model, and therefore is deemed to have been successfully replicated.
Statistical methods	Not involving regression or clustering algorithms. Already ruled out as part of literature review reflections.
Cluster-Based Best Match Scanning (CBMS)	Implemented CBMS by using Lyold's logic for Kmeans clustering and Pearson Correlation for the KNN regression. However, this solution cannot work for datasets of over 1 million rows, where it exceeds the maximum allowable number of values for the KNN regression. Therefore, CBMS is considered not suitable for Big Data.
Random hot deck imputations	Was not considered as has already been seen that it is applied upon small datasets from the literature review.
Clustering combined with feature selection and imputation	Overlaps partly with the CBMS algorithm in the sense that this method proposes 3 phases; clustering, imputation and feature selection. The imputation phase was applied with 2 algorithms, namely KNN and Multivariate Imputation by Chained Equations(MICE). The CRI version with KNN is the equivalent of the CBMS, which has been determined to be inadequate for Big Data. As MICE is not adequately supported by SKLearn library in python and available discussed libraries from literature such as 'impyute' were, the CRI algorithm will not be considered as part of our proposed methodology.
Extreme value analysis	This algorithm was applied on the features containing numerical data and found a certain percentage of outliers. However, the final vetting whether those outliers are errors would be based upon human judgement.
K means algorithm combined with TF-IDF	TF-IDF was applied upon the text data of two attributes. As no errors were visible, artificial errors were introduced, and following the application of k-means for only one cluster, it is noted that errors, which were less and therefore rarer in the attribute, were listed last in the cluster. Subsequently, a human expert needs to ascertain whether those last values are errors or acceptable outliers.

A. Evaluation of imputation algorithms

An important consideration to take into account for evaluating the algorithms implemented in the course of this research is the fact that because of the use of real life datasets, there is a lack of absolute certainty about what should be final corrected values. Secondly, with the 3 Vs' of Big Data, it is challenging to measure data quality levels using known existing metrics and measures Some evaluation metrics need the presence of 'truth' samples, which are considered as known correct values. Examples of such evaluation metrics are 'mean squared errors' or 'root mean squared errors'. Hence, those metrics were not used for evaluation of selected algorithms.

There are four types of imputation accuracy, as follows[18]:

(i) Predictive accuracy or effectiveness: maximum preservation of true values (of each imputed value); (ii) Ranking accuracy: maximum preservation of true ordering (ranks) relationship in imputed values; (iii) Distributional accuracy: maximum preservation of the distributions of true values; and (iv) Global estimation accuracy: maximum preservation of analytic results and conclusions.

Due to the inherent lack of true values in a real life Big Data scenario in the health industry, accurate evaluation of imputation is deemed to be impossible to be carried out following the above definitions. Hence, the **plausibility** of imputed values could be used as another evaluation criteria for algorithms concerned with imputation. This is effected largely with the use of statistical data editing, but also with **outlier detection** [19]. This research adopted the outlier value detection as a qualitative imputation evaluation method, since if ever some level of inaccuracy is induced after imputation, it will be detected by this evaluation approach. The actual outlier value detection was carried out using the z-score, which works perfectly to detect outliers amongst numerical data.

The following table provides a summary of differences in the amount of outliers detected in the original dataset (OD) and dataset with imputations (ID).

Table 4: Evaluation of ML algorithms

Algorithm	OD	ID	T	Conclusion
Linear regression algorithm	0	0	8.2	As no outliers have been introduced, this method is deemed to be plausible.
lp-norm regularisation (SRSp)	0	76340	10.8	As all the imputed values seems to have become outliers, this algorithm is deemed not plausible.

The logic is that if there are similar amounts of outliers for OD and ID, then the imputed values are considered plausible. The column T provides computation time in seconds; for this criteria also, the linear regression algorithm performs slightly better.

B. Outlier detection

Accuracy detection of numerical data was based upon the use of z-score, which is a well-known function for outlier detection. As there is no certainty over which values are errors without a 'truth' sample, there is the need to induce artificial errors. Evaluating z-score will not bring any new contribution to knowledge, hence this was not performed as part of this research. Around 20 errors were introduced in the 'Program Type' attribute to evaluate the error detection of text data. 15 of those errors were just dummy words such as ' hegfgf', and 4 were only single alphabets, such as 'b'. All the dummy words were highlighted as part of the clustering process but no single alphabet detected. This could be explained by the 'stop words' parameter set to English in the TF-IDF 'vectorizer' process. In any case, this would result in very high precision and recall benchmarks for inaccurate terms detection following the clustering. Another experiment was carried out using the ' Specialty' attribute, with the raw original data. The clustering process highlighted 75 distinct terms, and upon manual inspection, they were all correct English terms. Hence, they can be assumed to be accurate data. The final process of ascertaining whether a term is an error would rely upon human judgement. Using automated measures such as frequency and/or count of occurrences may show most probable errors, but an element of uncertainty would prevail which would still warrant human judgement as it is extremely challenging to distinguish correct and incorrect data values automatically, without a training process which had been explained to be inappropriate for Big Data in the accuracy DQD discussions of this study.

V. CONCLUSION

The overall aim of the paper was to investigate use of selected ML algorithms from existing literature to improve data quality operations, more specifically detection of data completeness and accuracy issues. The rationale for the use of ML algorithms stemmed from the characteristics of Big Data, namely volume, variety and velocity, which challenge data quality methods applicable in a non-Big Data context.

The literature review concluded that there were some previous uses of ML relative to the completeness DQD; however, most of the existing research studies involved ***imputation*** of missing values in the most effective way through ML, and not specifically the use of ML algorithms to ***detect*** missing values. Further investigation showed that the detection of missing values in a dataset is a straightforward process with tools such as RapidMiner Studio or programming languages such as python. Disguised Missing values have not been considered for this research study, since they may also be treated as outliers or inliers [20]. Consequently, this research objective diverted from its original aim of only detecting missing values, but carried out experiments to determine which ML algorithms could be most effective for imputation of missing values within Big Data in the health industry. The results concluded that the use of linear regression was better in terms of the plausibility of data values imputed and computation time of imputations. However, this result calls for further validation for the following reasons: 1) many ML algorithms discussed could not be implemented and therefore evaluated similarly, 2) some of the ML algorithms discussed, more specifically the clustering based algorithms, could not be applicable on datasets with over 1 million rows. Another challenge involved with the evaluation of the experiments occurred because the experiments involved a real world dataset, and therefore, the correct values for the missing ones were unknown. Due to this fact, the plausibility evaluation criteria was applied This dataset contained missing numerical values only, and hence future works might focus upon imputation of categorical values. Some interesting ML techniques, which have not been cited as part of the DQ literature, such as the use of Generative adversarial networks, should be investigated. The same text data outlier detection techniques could be applied as part of plausibility evaluation..

With regard to the use of ML algorithms to detect data inaccuracy issues, the first conclusion is that there is no unique ML algorithm that will be able to cater for all types of data inaccuracy issues. Inaccuracy issues as part of numerical data can be detected with non-ML algorithms such as the use of the statistical algorithm known as 'z-score' to detect outliers. Whereas for text data, a transformation of the text data into TF-IDF scores is required first, and then k-means (a clustering based ML algorithm) must be applied. The results showed that some artificially induced text errors were detected in this way. However, in terms of the accuracy DQD, an algorithm alone will not be sufficient to distinguish between data inaccuracy errors and genuine acceptable outliers. Hence, this research concludes that human expertise are needed to validate potential errors. Thus, a semi-automated approach is advised as part of data inaccuracy detection systems for Big Data in the health industry.

Overall, even if the literature review seemed to be pointing to regression and clustering ML algorithms as having the greater potential to solve data completeness and data inaccuracy issues for Big Data in the health industry, the experiments carried out as part of this research proved that it is very difficult to have an umbrella ML algorithm category capable of dealing with both types of DQ issues. In the case of missing value imputations, regression based algorithms tend to be more effective, both in terms of plausibility and computation time. Whereas in the case of data inaccuracy issues, specifically for text

data, clustering based ML algorithms may be more effective. Furthermore, some DQ issues such as detection of missing values and detection of outliers amongst numerical data do not specifically require the application of ML algorithms. This paper has demonstrated that supervised learning techniques can be relevant for supporting DQ activities; however, dividing an original dataset into training and test sections might be challenging with some types of data sources of Big Data, specially those affected by high velocity of data. Big Data characteristics also challenge evaluation of ML algorithms as typical evaluation benchmarks used in the ML domain, such as precision/recall scores and harmonic values, cannot be used in high volatility data sources. The absence of 'truth' values further exhacerbates challenges for evaluating ML algorithms.

A system aimed at improving DQ for Big Data in the health industry will need to devise a hybrid solution, mixing regression and clustering based ML algorithms with statistical tools for the detection of data incompleteness and inaccuracy issues. This solution should also be semi-automated, as the involvement of human expertise is deemed to be essential for detecting inaccuracy errors. The technologies used to develop the solution also will have a significant impact on the effectiveness of any subsequent data analytics undertaken. It is recommended that technologies allowing out-of-core computation to improve processing time for Big Data systems should be used.

REFERENCES

[1] H. Ma, I. King, and M.R. Lyu, "Effective Missing Data Prediction for Collaborative Filtering", in *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval.* ACM-Amsterdam, The Netherlands, 2007 p.39-46.

[2] S.García, S. Ramírez-Gallego, J. Luengo, J.MBenítez and F. Herrera "Big Data Preprocessing: Methods and Prospects", in *Big Data Analytics, Vol. 1, 9, 2016.*DOI:10.1186/s41044-016-0014-0 .

[3] W. Yu, , W. Zhu, , G. Liu,,B. Kan,, T. Zhao, and H. Liu, , . "Cluster-based Best Match Scanning for Large-Scale Missing Data Imputation", in *3rd International Conference on Big Data Computing and Communications*,2017. DOI:10.1109/BIGCOM.2017.48

[4] J. Ahmed and R. Soomrani,, "TDTD: Thyroid Disease Type Diagnostics". *2016 International Conference on Intelligent Systems Engineering(ICISE)*, 15 – 17 Jan, 2016, Islamabad, Pakistan.

[5] G. Rahman, , Z. Islam, ,T. Bossomaier, and J. Gao, ,.. "CAIRAD: A Co-appearance based Analysis for Incorrect Records and Attribute-values Detection", in *WCCI 2012 IEEE World Congress on Computational Intelligence* June, 10-15, 2012 - Brisbane, Australia.

[6] X. Chen ,Y. Cai, Q. Liu and L. Chen . "Nonconvex lp-Norm Regularized Sparse Self-Representation for Traffic Sensor Data Recovery", in *IEEE Access*, Vol 6, pp 24279-24290 DOI:10.1109/ACCESS.2018.2832043

[7] M.J. Loh, and T. Dasu . "Effect of Data Repair on Mining Network Streams*",* in *2012 IEEE 12th International Conference on Data Mining Workshops*.Dec 10 2012, Brussels, Belgium, ISBN: 978-1-4673-5164-5

[8] J. L. Schafer and J. W. Graham. "Missing Data: Our View of the State of the Art". In *Psychological Methods*, 7(2):147–177, 2002

[9] T. Lin, C. Yang and I. Chiang, "Improvement of Prognostic Models for ESRD Mortality by the Bootstrap Method with Random Hot Deck Imputation", in *2014 IEEE International Conference on Granular Computing (GrC)*, Noboribetsu, 2014, pp. 166-169.doi: 10.1109/GRC.2014.6982828

[10] C. Zhang, Y. Qin, X. Zhu, J. Zhang and S. Zhang, "Clustering-based Missing Value Imputation for Data Preprocessing," in *2006 4th IEEE International Conference on Industrial Informatics*, Singapore, 2006, pp. 1081-1086.doi: 10.1109/INDIN.2006.275767

[11] Cao Truong Tran. "Evolutionary Machine Learning for Classification with Incomplete Data" , PhD Dissertation, Victoria University of Wellington, New Zealand, 2018.

[12] N. Laranjeiro, S. Soydemir, and J. Bernardino. "A Survey on Data Quality: Classifying Poor Data", in *The 21st IEEE Pacific Rim International Symposium o n Dependable Computing (PRDC 2015), November 18-20,* Zhangjiajie, China 2015.

[13] D. Firmani,.,M. Mecella,,M.Scannapieco, and C. Batini,."On the Meaningfulness of 'Big Data Quality;", in *Data Science Engineering Journal*, 1(1):6–20, 2016.

[14] Y. Yuan, S.Li, X. Zhang and J. Sun . "A Comparative Analysis of SVN, Naïve Bayes and GBDT for Data Faults Detection in WSNs". 2018 IEEE International Conference on Software Quality, Reliability and Security Companion, 2018.

[15] M.S. JayaramHariharakrishnan, and S.K.B Sundhara Kumar. **"**Survey of Pre-processing Techniques for Mining Big Data", in *IEEE International Conference on Computer, Communication, and Signal Processing (ICCCSP-2017),* 10-11 January 2017, Chennai, India.

[16] X. Wu and X. Zhu. ''Mining with Noise Knowledge: Error-Aware Data Mining'', in *IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART A: SYSTEMS AND HUMANS*, VOL. 38, NO. 4, JULY 2008.

[17] H.Yinghao, , M. Yi Lu, and G. Yao. "Automotive Diagnosis Typo Correction Using Domain Knowledge and Machine Learning*",* in *2013 IEEE Symposium on Computational Intelligence and Data Mining (CIDM),* Singapore, 16-19 April, 2013

[18] J.M Pasteels,, "Review of best practice methodologies for imputing and harmonising data in cross-country datasets*",* ILO internal report, 2013.

[19] C. Sporleder,., M.V Erp,., T. Porcelijn,., and A.V Bosch, .. "Spotting The 'Odd-One-Out': Data-Driven Error Detection And Correction In Textual Databases", *Proceedings of the workshop on Adaptative Text Extraction and Mining(ATEM* 2006)

[20] A. Qahtan,, M. Ouzzani,, A. Elmagarmid, and N. Tang,.." FAHES: A Robust Disguised Missing Values Detector", Accessed on: August 24 2019.Available: https://www.researchgate.net/publication/325038265, 2018

[21] Gartner, "RapidMiner Reviews", Accessed on: 10 October 2020. Available: https://www.gartner.com/reviews/market/data-science-machine-learning-platforms/vendor/rapidminer/reviews