# Data analysis on big data applications with small samples and incomplete information

Soizic Linford
*CUE*
*Coventry University*
Coventry, UK
SLinford@cad.coventry.ac.uk

Benjamin Bogdanovic; Kuo-Ming Chao
*Institute of Future Transport and Cities*
*Coventry University*
Coventry, UK
bogdanob@coventry.ac.uk; k.chao@coventry.ac.uk

Slađana Janković; Vladislav Maraš; Mirjana Bugarinović
*Faculty of Transport and Traffic Engineering*
*University of Belgrade*
Belgrade, Serbia
s.jankovic; v.maras; mirab @sf.bg.ac.rs

Ilias Trochidis
*Ortelio Ltd*
Coventry, UK
it@ortelio.co.uk

***Abstract*—The EU and other public organizations at different levels of national and local government across the world have funded and invested in numerous research and development projects on big data transport applications over last few years. The mid and long term effectiveness of these applications is very difficult to measure, and the benefits and usability of these applications are not easy to calculate. NOESIS, funded under EU H2020 program, aims to design a decision supported tool by gathering and analyzing these applications as use cases to formulate sufficient knowledge for policy makers to make informed decisions for their big data transport applications.**

***The challenges in this work are associated with a small number of samples, with incomplete information, but having a good size of features that need to be analyzed to make a confident enough recommendation. This paper reports various statistical and machine learning approaches used to address these challenges and their results.***

**Keywords—big data, machine learning, random forest, multivariate regression**

## I. Introduction

NOESIS is an EU funded project that looks to build a library of big data in transportation use cases for analysis and evaluation to understand the landscape and impact of big data in transportation. The project aims to formulate an analysis outcome, develop a learning framework, and create a value capture mechanism, which would provide decision and policy makers with the expected benefits and costs from their estimates. The expected impact of the project is to identify critical factors and features that lead to successful implementation of big data technologies and services in the field of transport and logistics, with significant value generation from a socioeconomic viewpoint.

Two rounds of information retrieval to extract the features and factors are been carried out. The first round identifies and gathers important knowledge from a group of experts after several meetings and discussions, leading to a list of questions that are distributed to the practitioners who have conducted big data transportation projects to answer. The answers and questions become use cases which are stored in the library for the second round of analysis to determine their correlations. A use case questionnaire including 65 questions that are considerably important factors and features for big data transportation is generated. They are grouped into 6 categories and these are; General information; Transport-related information; Transport challenges faced with the use case; Data-related information; Privacy, security and governance information; and Value creation. The questions contain a mixture of open and closed, with possible answers in text or numeric format. At this stage, we collect 52 case uses with inconsistent qualities, diverse transportation sectors, various investment sizes, and different stages of their projects.

We are facing a data set which has a high-dimension, high-diversity, low-volume, and mixture of structured/unstructured data with missing values. This paper reports the application of statistical and machine learning methods to address these challenges and present the outcomes and lessons learned.

## II. Data Science Pipeline

There are no standardized steps and frameworks for the data science pipeline, which is a critical process determining success of machine learning application on big data [1]. These steps are varied often depending on the application domains and data properties that drive and shape the procedures and learning models to meet their requirements. However, each step in the pipeline, from raw to final data set, involves a series of decision making which has impact on the final data and knowledge quality. In other words, each decision at every step could introduce bias or create inaccuracies along the process. For example, the threshold for data clearing is to remove noise data or outliers, but the decision could be subjective and rule out knowledge or insights that can be useful in some scenarios or cases. The chosen algorithm is also likely to incorporate some biases [2]. Iterative experiments and analysis can reduce biases and gain better insight of data and knowledge.

The data science pipeline (see Fig. 1) for this exercise starts with the question “What data needs to be collected”; then collects data; filters data; transforms data format; normalizes data; interpolates data; choose learning and statistical methods to train and test; examine the results, if not satisfactory; determining key attributes; determine multiple outputs or single output; start the next round of analysis until satisfactory results are produced or termination conditions met.

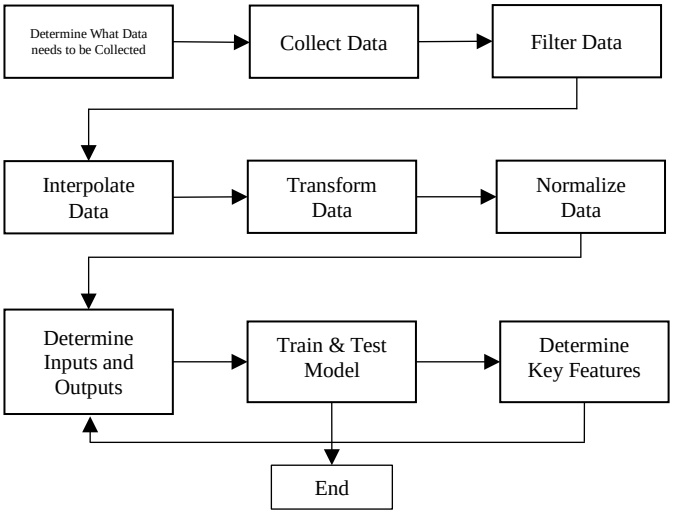

Fig 1. Data Science Pipeline

The questions are designed to gather the nature of big data projects in transport, background information, adopted technologies, challenges they want to address, and investment return benefits. These questions can be considered as variables, attributes or features that can be classified into dependent, independent, and explainable ones. For example, the answer for questions about short project descriptions and big data solutions are open and free form text, that makes them difficult to analyze, but the answer could be annotated to mediate this (see Table I). The question, "Big Data service/solution" can have a multiple answers, rather than single one, which results in the choice of converting between multiple attributes, and cardinality of a single attribute. Few questions fall into this category. "Business Section and type" and "Size of the organization" etc. are independent attributes and the questions e.g. "Yearly Operating Expenses (OPEX)", "life-cycle costs savings" etc. in the value creation are dependable attributes, providing important information in the supervised learning process to evaluate the quality of learning mechanism (see Table I). The aim for the learning mechanism is to identify the relationship between dependent and independent attributes, so users can obtain the estimated answer of dependable attributes by giving values to independent attributes. Gaining answers to these questions however, is not a trivial task due to maturity of use cases, and data confidentiality and sensitivity etc.

The process of data filtering is used to remove low quality use cases or attributes which have over 60% missing values or irrelevances. Since the learning mechanism or statistical methods can only take numerical values, the textual values are transformed to numbers as coherently or meaningfully as possible. For example, Number 1, 2, and 3 was adopted to represent low, medium and high respectively for questions like, "To better know clients/users". The values given to questions which could have multiple answers, for example, "application areas", are translated to the aggregate of the items in answer, instead of having each separate item for a column, reducing complexity. Once all attributes have numeric values, the normalization process takes place to reduce the gaps among these numbers to avoid skewing the results. The application of linear interpolation approach fills missing values in the data set ready for the chosen learning mechanisms and statistical methods to model.

TABLE I. EXAMPLES OF NOESIS USE CASE CATEGORIES AND ATTRIBUTES

| **General information** | **Value creation** |
|---|---|
| Use case title | CAPITAL EXPENDITURES (CAPEX) |
| Use case ID | Financing of CAPEX |
| Short description | Yearly OPERATING EXPENSES (OPEX) |
| Big Data application(s) | Financing of OPEX |
| Big Data service/solution | Final clients |
| Stakeholders | Job creation |
| Organization type | Other benefits for the society |
| Business sector | Benefit/CAPEX-OPEX ratio |

The nature of the problem domain is to model the data for prediction rather than classification. In addition, the project only has a small rather than large data set, with several missing values. A Random Forest (RF) method [3] was selected to train the model, as it combines the benefits of bagging and decision trees, with the capability of dealing with the applications having large attributes in a small data set [3]. A statistical multiple variate regression approach is adopted to cross-validate the results. The learning process needs at least two subsets of data, one for training, and the other for testing. Since the data set in this instance is small, 10% of 20% of the data is designated for testing. The problem domain can be characterized using multivariate regressions, which could optimize single and multiple outputs with the whole set of input/independent attributes and some selected ones.

Several feature (attribute) selection approaches [4] can be applied to identify the significance of attributes to the outputs. We start with the mechanism provided in the RF to calculate the attribute significances while training the models. Ten of the most significant independent attributes are selected for model refinement. The whole process ends when a satisfactory result is produced, or it seems no further improvements on the results can be achieved.

## III. MULTIPLE VARIATE REGRESSION AND FEATURES

Multivariate regression models with multiple inputs/ independent attributes can predict single output or multiple outputs, or dependent attributes [5].

If the application requires multiple outputs, it can be modelled separately for each target and aggregate these individual outputs to one single multi-output. Each model in this case was trained to optimise each single output, rather than all the targets together, so the collection of their outputs could not reflect the correlation between these dependent attributes. It could be an expensive computational and time-consuming exercise due to separate training resources required for each individual variable. It would be more expensive if additional algorithms are adopted to calculate the significances of independent attributes for each output before the training. Even if the outcome is satisfactory, it could still be a challenge to

analyse the relationships among the independent and dependent attributes and their values.

Modelling multivariate regression for one output has the advantages of being able to reduce the search space, and providing better justifications of the relationships when there are too many attributes involved and missing values in the data.

In a multivariate regression model, feature (attributes) selection method is to determine importance of features and to remove less important ones before the training. It is aimed to improve learning accuracy, interpretability and computational performance, by eliminating noisy, irrelevant, and redundant attributes and data. It searches all possible combinations of subsets of independent attributes from the whole set of attributes and then evaluates their contribution to output.

### A. *Feature Selection*

Based on the different strategies of searching, feature selection methods can be filter, wrapper, or embedded [6]. Filter methods evaluate, and rank attributes based on certain criteria, and select the most significant ones before training (e.g eigenvectors). The subset is generated and selected according to eigenvectors which indicate the degree to which the reduced data that can be restored back to the original space. Wrapper methods evaluate and cross-validate to select the features based on the selected learning algorithm (e.g. Genetic Algorithms). Embedded models are like Wrapper ones, but they perform feature selection in the process of model construction (e.g. Decision Trees and RF) [6].

Weka (Waikato Environment for Knowledge Analysis) [7], an open source software, supports Filter and Wrapper methods. SKlearn, [8] a publicly available platform with a Python based library for developing big data analytics, provides an Embedded method. Both support several main machine learning and statistical methods including Random Forest, Supported Vector Machine (SVM), and Multivariate Linear Regression functions, for data mining and modelling.

Weka is used to select features or attributes, before the data feeds to the machine learning methods [9] in SKlearn. We use its RF function as the main tool to model our data set, and MR was used to validate the results, which is run in parallel with RF. SVM, an additional machine learning method, is used to verify the result, as it requires less parameters.

In the Weka attribute selection, the task was divided into single-attribute evaluation and attribute subsets evaluation. In the single-attribute evaluation, two methods, correlationAttributeEval evaluator/Ranker search, and GainRatioAttributeEval evaluator/Ranker search, were used to explore the most significant independent attributes of a single target. Experiments using attribute subsets evaluation methods based on CfsSubsetEval (Correlation-based Feature Subset Selection Evaluator) evaluator/BestFirst search, and WrapperSubsetEval evaluator/BestFirst search, were carried out to find a set of most important attributes. The results of the most important attributes derived from these methods can be found in Table II.

TABLE II. FEATURE SELECTIONS BASED ON WEKA

| CorrelationAttributeEval evaluator & Ranker search method | GainRatioAttributeEval evaluator & Ranker search method | CfsSubsetEval evaluator & BestFirst search method | WrapperSubsetEval evaluator & BestFirst search method |
|---|---|---|---|
| business_processes | data_analysis_method | data_analysis_method | data_variety |
| data_variety | data_variety | data_variety | variable_sources |
| structure_for_managing_projects | data_processing_technique_tool | data_processing_technique_tool | sector |
| long_term_vision | variable_sources | data_velocity | data_veracity |
| variable_sources | data_veracity | variable_sources | |
| | data_source | data_source | |
| | | current_data_openness_level | |

Table III illustrates the relationship between the independent selected attributes and the individual target attribute (i.e. application_period and benefit_capex_opex_ratio), the number of subsets being explored and evaluated (i.e 166 and 132), and the merits of best subsets evaluated (i.e. 0.577 and 0.559). The higher the merit score, the better. From the number of subsets being evaluated, it seems the search space has been reasonably explored and exploited. Their merit scores are just over 0.5 which are not significant, considering its range is between 0 and 1. Their true effects will be examined when these are fed to RF and MR to generate predictions.

TABLE III. WEKA SELECTED ATTRIBUTES

| Class (nominal) | benefit_capex_opex_ratio | application_period |
|---|---|---|
| Subsets evaluated | 132 | 166 |
| Merit of best subset found | 0.559 | 0.577 |
| Selected attributes | data_analysis_method, data_processing_technique_tool, data_variety | data_analysis_method, data_source, data_variety, data_velocity, variable_sources, business_processes |

### B. *Random Forest*

The Random forest is an ensemble learning method which can conduct regression or classification tasks in data analysis with the same type of multiple decision trees. The random forest algorithm is known for its capability of working scenarios that have missing values, have not scaled well, and are bias, due to the joint contribution of a large number of decision trees [3].

The performance of the algorithm depends on the setting of a number of key parameters; the maximum number of trees, number of features to generate the branch, maximum levels in

decision tree, minimum number of data points to branch out, minimum number of data points in a leaf node, and the use of bootstrapping methods for sampling data points (with or without replacement). The algorithm uses the above parameters to build trees to form a forest. Each tree is built by randomly selecting a subset of data and features. There are no explicit guidelines to set values, so adjusting them after few trials may be required. The data set is divided into two subsets for the purposes of training and testing.

The algorithm begins by selecting random records from the training dataset to build trees according to the given parameters. The process iterates until the stop condition has met, which is either the specified number of trees being built, or all the data being analyzed.

Each tree in the forest produces a prediction value(s), and their values could be different, so an agreed result can be derived from a most common approach, by averaging all values generated from all the trees in the forest. Different ways (e.g. weighted average) of calculating the final prediction could be applied depending on the applications. Due to its group contribution by considering different aspects, RF has advantages in reducing bias, tolerating of missing values, and being adaptable to small data set sizes. RF does not only produce regression results, but also the significance of the list of attributes contributing to the result.

## IV. Experments and Analysis

The designed RF, MR, and feature selection methods are implemented in Python and Java with libraries imported from SKlearn [8] and Weka [7] to train the models on the data set. The experiments on RF for multiple independent attributes and multiple outputs were carried out to evaluate the model performance, and to identify the most significant independent attributes to the results. RF uses the selected highest significant attributes to predict multiple output attributes to examine any improvements. MR is also employed to model the same data sets and attributes to examine any deviations from RF. The results are evaluated using different ways of calculating error rates; Mean Absolute Error (MAE), Mean Absolute Error (MAE), and Root Mean Squared Error (RMSE). They provide a clear and direct indication on the learning performance.

Table IV shows the error rates of RF and MR having optimized all attributes of the data set, which have 21 independent ones and 32 dependent ones. The normalized value range is between 0 to 10 across all attributes, but the values of some attributes fall only in the range between 0 to 3. Therefore, these aggregated error values are relatively small, and provide some insights on the learning performance, but they are still difficult to interpret.

Table V lists the sorted independent attributes in descending order, and their weighted contribution to the result calculated by RF during training. The resulting top 10 most significant attributes are highlighted in bold.

Figure 1 shows a bar chart depicting the significance of each attribute to the result of the RF, and it also indicates that no single attribute has dominant influence on the prediction.

TABLE IV. The modelling results of RF and MR

| Error Rate | RF | MR |
|---|---|---|
| Mean Absolute Error | 0.59 | 0.62 |
| Mean Squared Error: | 0.71 | 0.78 |
| Root Mean Squared Error: | 0.84 | 0.88 |

TABLE V. Features and their Significances to the result

| Feature No | Feature Name | |
|---|---|---|
| 8 | **data_processing_technique_tool** | 0.091182 |
| 17 | **current_data_openness_level** | 0.085975 |
| 12 | **data_veracity** | 0.083483 |
| 15 | **current_personal_privacy_concern_level** | 0.068402 |
| 16 | **business_confidentiality_problem_level** | 0.066210 |
| 11 | **data_velocity** | 0.065985 |
| 19 | **framework_for_privacy_issues** | 0.054819 |
| 18 | **business_processes** | 0.053131 |
| 7 | **data_analysis_method** | 0.050090 |
| 2 | **organization_type** | 0.043917 |
| 1 | business_sector | 0.045494 |
| 0 | Application | 0.043425 |
| 14 | variable_sources | 0.038874 |
| 20 | long_term_vision | 0.035327 |
| 9 | data_source | 0.031172 |
| 10 | data_variety | 0.029278 |
| 6 | time_horizon_until_implementation | 0.029560 |
| 5 | study_area | 0.026942 |
| 13 | existing_data_gap | 0.022757 |
| 4 | Sector | 0.018704 |
| 3 | Modes | 0.015744 |

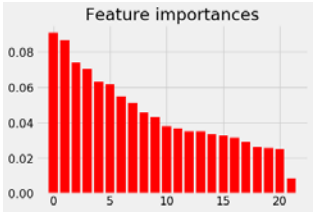


Fig. 1. Feature importance to the result

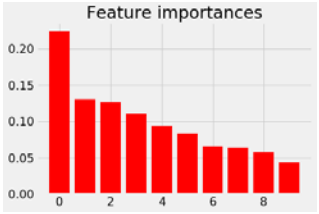


Fig. 2. Top 10 most important features to the result

The following table (Table VI) shows the Precision scores of individual attributes which have a mixed picture of good and poor results in the column “Precision: All independent attributes”. The precision is based on the number of True Positives divided by the sum of True Positives and False Positives. The prediction values are translated in to classes by rounding floating numbers to integers, such as 1.2 to 1 or 2.6 to 3, to obtain the precision rate.

In general, the model makes a better prediction on finance related attributes, but not so well on data related and certain technical issues (e.g. QoS, safety and security). Several predictions on attributes ending with 0.5 Precision score show non-conclusive outcomes.

The following experiment attempts to remove those with less weighted attributes, with only top 10 weighted independent

attributes based on RF (see Table VII) being used to predict the whole of dependent attributes. Table VII lists the top 10 most significant attributes generated by RF and their significance to the result. It shows there is no dominant influential attribute to the result. Figure 2 displays the ranking of the most influential independent attributes which have a different order than shown in Figure 1. Its outcomes are summarized in Table VI and X, and it also reveals that there is not much improvement on the Precision scores compared with the first experiment result (see the column "Precision Score: Top 10 independent attributes" in Table VI), and in the error rates (see Table X). In some cases, the Precision score in the simplified version is worse than those with a full set of independent attributes. The MR does not perform well in the case of reduced attributes either. Table IX illustrates the error rates of RF and MR, values of which seem acceptable, as they are marginally better than the results using the whole set of independent attributes.

TABLE VI. PRECISION RATES FROM RF WITH ALL AND SELECTED FEATURES

| Dependent Features / Attributes | Precision: All independent attributes | Precision Score: Top 10 independent attributes |
|---|---|---|
| application_period | 0.33 | 05 |
| automation | 0.67 | 0.5 |
| data_related | 0.17 | 0.17 |
| environment_and_health | 0.67 | 0.67 |
| freight_and_logistics | 0.5 | 0.5 |
| integration | 0.67 | 0.67 |
| maintenance | 0.5 | 0.5 |
| quality_of_service | 0.33 | 0.33 |
| resilience | 0.5 | 0.5 |
| safety_and_security | 0.33 | 0.17 |
| capex | 0.67 | 0.5 |
| transport_management_and_operation | 0.5 | 0.67 |
| transport_policy_and_planning | 0.67 | 0.67 |
| final_clients | 0.33 | 0.5 |
| financing_of_capex | 0.67 | 0.67 |
| financing_of_opex | 0.83 | 1 |
| improve_know_how | 0.5 | 0.33 |
| job_creation | 0.83 | 0.67 |
| life_cycle_costs_savings | 0.67 | 0.5 |
| more_revenue | 0.5 | 0.5 |
| new_business_opportunities | 0.5 | 0.5 |
| opex | 1.0 | 1.0 |
| to_better_know_clients | 0.83 | 0.83 |
| to_improve_coordination_across_stakeholders | 0.5 | 0.34 |
| to_improve_information_provision | 0.5 | 0.67 |
| to_improve_quality_of_service | 0.5 | 0.5 |
| to_improve_reliability | 0.5 | 0.67 |
| to_improve_safety | 0.5 | 0.5 |
| to_improve_supervision | 0.5 | 0.5 |
| to_reduce_environmental_impacts | 0.5 | 0.5 |
| travel_time_savings | 0.5 | 0.5 |

The above findings do not only consider the relationship among the independent and dependents, but also the interdependency among the dependents. The prediction outcomes were not as precise expected, as only 11 out of 31 target attributes have more than 0.5 scores (see the highlighted values in Table VI). In other words, these predictions are better than random guesses. Predictions on five attributes which have less 0.5 precision rate are not reliable, as the trained model has more incorrect predictions than correct ones.

More experiments use MR and MF to optimize multivariate input attributes, but one single output or dependent variable, to examine if the multivariate approach can perform better on individual dependable variable rather than multiple ones.

TABLE VII. TOP 10 WEIGHTED INDEPENDENT ATTRIBUTES BASED ON RF

| | | |
|---|---|---|
| 0 | business_processes | 0.223919 |
| 1 | business_confidentiality_problem_level | 0.129835 |
| 2 | data_processing_technique_tool | 0.126602 |
| 3 | current_data_openness_level | 0.109958 |
| 4 | current_personal_privacy_concern_level | 0.094354 |
| 5 | framework_for_privacy_issues | 0.083410 |
| 6 | data_veracity | 0.066275 |
| 7 | organization_type | 0.063499 |
| 8 | data_velocity | 0.058048 |
| 9 | data_analysis_method | 0.044099 |

TABLE VIII. THE RESULT OF TOP 10 INDEPENDENT ATTRIBUTES BASED ON RF

| Error Rate | RF | MR |
|---|---|---|
| Mean Absolute Error | 0.59 | 0.60 |
| Mean Squared Error: | 0.71 | 0.74 |
| Root Mean Squared Error: | 0.84 | 0.86 |

Table IX shows the top 10 most weighted features evaluated by Weka CorrelationAttributeEval and Ranker search methods for application_period class to predict the target attribute. RF and MR use these selected attributes to model the data set to predict 32 attributes separately. The list of attributes in the table, with the weights calculated by RF, have a different order from its original one evaluated by Weka. Different sets of features for each individual target attributes selected by the other three evaluators and search methods supported Weka are also used by RF and MR. Due to space limitations, only three output attributes and two feature selection methods (WrapperSubsetEval and BestFirst search method (WB) and CorrelationAttributeEval and Ranker (CR)) [7,9] are sampled, as they are the most representative.

TABLE IX. TOP 10 HIGHEST WEIGHTED FEATURES EVALUATED BY WEKA FOR APLICATION_PERIOD

| Feature No | Feature Name | Significance |
|---|---|---|
| 9 | long_term_vision | 0.222484 |
| 2 | data_processing_technique_tool | 0.196759 |
| 5 | data_velocity | 0.104494 |
| 4 | data_variety | 0.088264 |
| 8 | variable_sources | 0.082806 |
| 1 | data_analysis_method | 0.071767 |
| 6 | data_veracity | 0.071221 |
| 0 | organization_type | 0.063782 |
| 3 | data_source | 0.049588 |
| 7 | existing_data_gap | 0.048834 |

For the target attribute application_period shown in Table X, RF applies three sets of selected features evaluated by three different approaches to optimize the data set. It obtains the same

precision rate, even though RF10 has a lower error rate than other two.

TABLE X. THE REUSLT FOR APLICATION_PERIOD

| | Weka (CR) | RF 10 | Weka (WB) |
|---|---|---|---|
| MAE/RF | 0.77 | 0.72 | 0.93 |
| MSE /RF | 1 | 1 | 1.41 |
| RMSE/RF | 1 | 1 | 1.19 |
| RMAE/MR | 0.77 | 0.72 | 0.94 |
| RMSE/MR | 1 | 1 | 1.41 |
| RRMSE/MR | 1 | 1 | 1.19 |
| Precision | 0.5 | 0.5 | 0.5 |

TABLE XI. THE REUSLT FOR OPEX

| | Weka (CR) | RF 10 | Weka (WB) |
|---|---|---|---|
| MAE/RF | 0.19 | 0.16 | 0.21 |
| MSE /RF | 0.15 | 0.10 | 0.14 |
| RMSE/RF | 0.38 | 0.32 | 0.37 |
| RMAE/MR | 0.19 | 0.16 | 0.20 |
| RMSE/MR | 0.15 | 0.10 | 0.13 |
| RRMSE/MR | 0.38 | 0.32 | 0.37 |
| Precision | 0.83 | 0.83 | 0.83 |

The result of predicting OPEX is summarized in Table XI. RF produces a high precision rate in prediction for OPEN with three sets of selected attributes which have slightly different error rates, but with the same precision rate.

TABLE XII. THE REUSLT FOR FINAL_CLIENTS

| | Weka (CR) | RF 10 | Weka (WB) |
|---|---|---|---|
| MAE/RF | 1.30 | 1.35 | 1.25 |
| MSE /RF | 1.65 | 3.07 | 2.63 |
| RMSE/RF | 1.54 | 1.75 | 1.62 |
| RMAE/MR | 1.30 | 1.35 | 1.25 |
| RMSE/MR | 2.73 | 3.07 | 2.63 |
| RRMSE/MR | 1.65 | 1.75 | 1.62 |
| Precision | 0.33 | 0.5 | 0.5 |

Table XII shows the results of RF modeling Final_Client with three sets of selected independent attributes. Weka (WB) has better performance than the other two in terms of error rate, but its precision rate remains the same as RF 10.

The results show there are no significant differences in prediction among the selected features or attributes which are generated from Weka evaluation or RF methods for RF and MR. The multivariate prediction for single output has produced slightly better results than those for multiple outputs. SVM produces worse results than RF and MR.

## V. CONCLUSION AND FUTURE WORK

In this paper, we report an attempt of employing machine learning and statistical methods to analyze a small data set, which has 65 attributes with several missing values transformed from transport big data use cases, leading to the design of a decision supported system for investment. RF and MR were adopted to train the model on the data, and several feature selection methods supported by Weka were also used to remove insignificant attributes in the training, with intention of reducing the complexity and increase the predication accuracy. The evaluation of the trained model is carried out at a test phase by examining the values of MAE, MSE, RMSE, and checking the precision using a set of new use cases.

The results show that RF and MR can model the data set with or without feature selection, and produce consistent precise predictions on some attributes that can be used to support decisions. The results also evidently demonstrate that RF and MR, in this case, can only predict 11 out of 31 attributes with better precision rates than random in multiple target or output regression, despite using all or just the top 10 most significant independent input/attributes in modeling. Little improvement on the quality of the prediction has been found when both models only use small set independent attributes, derived from feature selection methods provided in Weka and RF to predict one single output. The results suggest that the performance of machine learning and statistical methods still have room for improvement when optimizing large numbers of features, with a small data set, and several missing values.

Due to the number of attributes or features involved in use cases, extrapolating a larger set of data from small samples, or expansion from the synthesised data, requires careful design to avoid result distortion and skew. This task is on our list of future work. We will continue to collect more use cases to further study the impact of missing values on RF modelling and other machine learning methods. In light of a recent report on material defection prediction [10], further research on using deep learning to model relatively small sets of data with large number of attributes will be carried out.

## ACKNOWLEDGMENT

The research, NOESIS project, is partially funded by EU program H2020, under Grant agreement no 769980. We also would like to acknowledge the efforts from the project team.

## REFERENCES

[1] C. Phethean, E. Simperl, T. Tiropanis, R. Tinati, and W. Hall, "The Role of Data Science in Web Science", IEEE Intelligent Systems, vol 31, no. 3, 2016, pp. 102– 107.

[2] W.M.P. Aalst, M. Bichler, and A. Heinzl, "Responsible Data Science", Bus Inf Syst Eng vol. 59, no. 311, https://doi.org/10.1007/s12599-017-0487-z, 2017.

[3] L. Breiman, "Random Forests", Machine Learning, vol. 45, no 5. https://doi.org/10.1023/A:1010933404324, 2001.

[4] J. Cai, J. Luo, S. Wang, and S. Yang , "Feature Selection in Machine Learning: A new perspective, Neurocomputing, vol. 300, 2018, pp. 70-79

[5] H. Borchani et al. "A survey on multi-output regression", Wiley Interdiscip. Rev. Data Min. Knowl. Discov.vol.5, 2015, pp. 216-233.

[6] J. Miao, and L. Niu, "A Survey on Feature Selection", Procedia Computer Science, vol. 91, 2016, pp. 919 – 926.

[7] R. Bouckaert, E. Frank, M. Hall, R. Kirkby, R. Reutemann, A. Seewald, and D. Scuse, "WEKA Manual for Version 3-8-3", The University of Waikato, Hamilton, New Zealand, https://netix.dl.sourceforge.net/project/weka/documentation/3.8.x/Weka Manual-3-8-3.pdf, 2018

[8] F. Pedregosa *et al* Scikit-learn: Machine Learning in Python, JMLR, vol. 12, pp. 2825-2830, 2011.

[9] Witten, E. Frank, M. Hall, and C. Pal, Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations, Morgan Kaufmann Publishers, 2016.

[10] S. Feng, H. Zhou, and H Dong, "Using deep neural network with small dataset to predict material defects", Materials & Design, vol. 62, pp. 300-310, 2019.