Power to the People!
Meta-Algorithmic Modelling in Applied Data Science

Marco Spruit and Raj Jagesar
Information and Computing Sciences, Utrecht University, Princetonplein 5, Utrecht, The Netherlands

Keywords: Applied Data Science, Meta-algorithmic Modelling, Machine Learning, Big Data.

Abstract: This position paper first defines the research field of applied data science at the intersection of domain expertise, data mining, and engineering capabilities, with particular attention to analytical applications. We then propose a meta-algorithmic approach for applied data science with societal impact based on activity recipes. Our people-centred motto from an applied data science perspective translates to design science research which focuses on empowering domain experts to sensibly apply data mining techniques through prototypical software implementations supported by meta-algorithmic recipes.

1 APPLIED DATA SCIENCE

Pritzker and May (2015:7) define Data Science as "the extraction of actionable knowledge directly from data through a process of discovery, or hypothesis formulation and hypothesis testing". In addition, they also relate the skills needed in Data Science. Based on their observations we propose to define Applied Data Science as follows:

> *Applied Data Science (ADS) is the knowledge discovery process in which analytical applications are designed and evaluated to improve the daily practices of domain experts.*

Note that this is in contrast to fundamental data science which aims to develop novel statistical and machine learning *techniques* for performing Data Science. In Applied Data Science the objective is to develop novel analytical *applications* to improve the real world around us. From the perspective of the Data Science Venn diagramme (Pritzker and May, 2015:9), Applied Data Science focuses on the *Analytical applications* intersection between the Domain expertise and Engineering capabilities. Finally, we observe an analogy with the ubiquitous people-process-technology model where technology aligns with machine learning algorithms, organisational processes are operationalised through analytical software implementations, and domain expertise is captured from, and enriched for, skilled professionals. Hence the motto: *power to the people!* Figure 1 contextualises the research field of and needed skills in Applied Data Science.

It is from this novel Applied Data Science research perspective that we investigate the core data science topic of machine learning in the remainder of this paper, from a meta-algorithmic modelling approach.

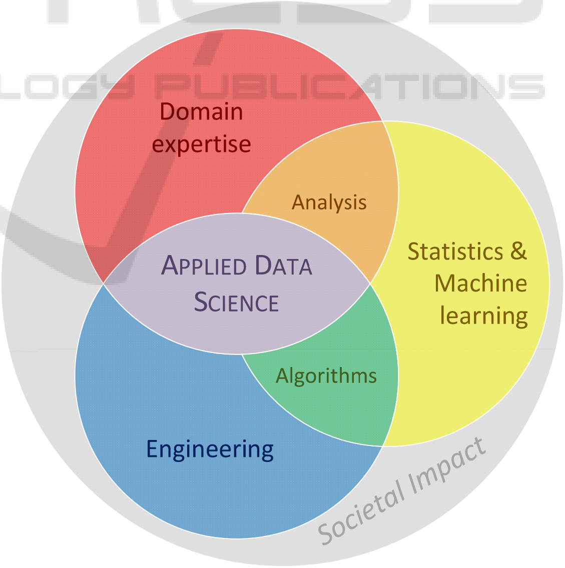

Figure 1: Applied Data Science in context.

2 MACHINE LEARNING

With the steadily growing availability of data storage space and computing power, advanced data

Spruit, M. and Jagesar, R.
Power to the People! - Meta-Algorithmic Modelling in Applied Data Science.
DOI: 10.5220/0006081604000406
In *Proceedings of the 8th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2016) - Volume 1: KDIR*, pages 400-406
ISBN: 978-989-758-203-5

mining efforts are coming within reach of increasingly more people. One common approach to perform a data mining project, and central to this ADS type of research, is to apply Machine Learning (ML) techniques. The application of ML techniques spans various disciplines like mathematics, statistics and computer science. These disciplines combined support the act of learning and result in models that are fitted to data. The challenge is to derive models that are accurate in the sense that they reflect the underlying patterns in the data whilst ignoring peculiarities that do not represent reality. A popular and well known purpose of these models is to make predictions on new and unseen examples of data. However, ML techniques are also well suited to explore the underlying patterns of a dataset. More often than not, machine learning techniques are employed to learn about the structure of a data set (Hall *et al.*, 2011). ML as a research field can be considered to be positioned at the heart of fundamental data science, as it requires both data mining and engineering expertise. This is also reflected in Figure 1 (Algorithms, in green colour).

3 PROBLEM STATEMENT

However, despite the growing usage and popularity of machine learning techniques in data mining projects, correctly applying these techniques remains quite a challenge. We list the three main challenges below:

1. *Depth versus breadth*: The ML field knows many different use cases, each of which has a sizeable body of literature surrounding the specific cases. The literature is usually found to be heavy on mathematical terminology and aimed at the computer science community. This prevents researchers from other fields in learning and correctly applying machine learning techniques in their own research (Domingos, 2012).
2. *Selection versus configuration*: In line with the aforementioned, applying machine learning techniques confronts users with many degrees of freedom in how to assemble and configure a learning system. One example of this is the fact that algorithm performance is largely determined by parameter settings, these settings are specific for each class of algorithm. However, in practice end users usually do not have enough knowledge on how to find optimal parameter settings (Yoo *et al.*, 2012). Many users leave the parameters to their default settings and base algorithm selection on reputation and / or intuitive appeal (Thornton *et al.*, 2013). This may lead to researchers using underperforming algorithms and gaining suboptimal results.
3. *Accuracy versus transparency*: Concerning the creation of models: ML shows that currently there is a trade-off to be had between accuracy and transparency (Kamwa *et al.*, 2012). In practice this means that algorithms which yield a high amount of insight into the data do not perform as well as their non-transparent (black box) counterparts and the other way around.

In order to get a better grip on these challenges, we propose a meta-algorithmic modelling approach, which we define as follows:

> *Meta-Algorithmic Modelling (MAM) is an engineering discipline where sequences of algorithm selection and configuration activities are specified deterministically for performing analytical tasks based on problem-specific data input characteristics and process preferences.*

MAM as a discipline is inspired by Method Engineering, “the engineering discipline to design, construct and adapt methods, techniques and tools for the development of information systems” (Brinkkemper, 1996). In related work, Simke (2013) describes a reusable, broadly-applicable set of design patterns to empower intelligent system architects. Finally, MAM also conceptually resembles the Theory of Inventive Problem Solving (TRIZ), a method for creative design thinking and real problem solving, partly due to its “Meta-Algorithm of Invention” (Orloff, 2016).

The strategic goal of MAM is to provide highly understandable and deterministic method fragments —*i.e.* activity recipes—to guide application domain experts without in-depth ML expertise step-by-step through an optimized ML process following Vleugel *et al.* (2010) and Pachidi and Spruit (2015), among others, based on the Design Science Research approach (Hevner *et al.*, 2004). We thereby promote reuse of state-of-the-art ML knowledge and best practices in the appropriate application of ML techniques, whilst at the same time provide information on how to cope with challenges like parameter optimization and model transparency (Pachidi *et al.*, 2014).

We argue that this MAM approach aligns especially well with the Applied Data Science perspective which we pursue in this research.

4 RESEARCH APPROACH

By taking into account our problem statement context above the overarching research question is formulated as follows:

> *How can meta-algorithmic modelling as a domain independent approach in an applied data science context be operationalised to guide the process of constructing transparent machine learning models for possible use by application domain experts?*

We will initially proceed with a limited scope: the creation of method fragments focused on supervised machine learning for binary classification tasks on structured data. This type of machine learning is concerned with deriving models from (training) data that are already available. Coincidentally this is one of the most applied and mature areas within the machine learning practice (Kotsiantis *et al.*, 2007).

First a theoretical foundation is established on the subjects of data mining, machine learning and model transparency. The concepts derived from this foundation are then grouped using the structure of a data mining process model. For our purposes we apply the base structure of the CRISP-DM process model and group the concepts into the following phases: data understanding, data preparation, and modelling & evaluation. Our method fragments will be composed using the same structure.

In this work we employ method engineering fragments notation to specify the meta-algorithmic models. More specifically, we apply the meta-modelling approach which yields a process-deliverable diagram (PDD; Weerd *et al.*, 2008). A PDD consists of two diagrams: the left-hand side shows an UML activity diagram (processes) and the right-hand side shows an UML class diagram (concepts or deliverables). Both diagrams are integrated and display how the activities are tied to each deliverable. Lastly, the activities and the concepts are each explained in separate tables. However, due to page restrictions these explanatory tables are excluded from this paper.

5 MODEL TRANSPARENCY

The concept of model transparency occasionally surfaces in the body of literature. In particular, when it concerns decision support systems where it must be clear how a system came to a certain (classification) decision (Johansson *et al.*, 2004; Olson *et al.*, 2012; Kamwa *et al.*, 2012b; Allahyari *et al.*, 2011).

There is consensus in the literature about the types of algorithms that are known to yield transparent and non-transparent (black box) models. Both tree and rule models are considered as transparent and highly interpretable. On the other hand, artificial neural networks, support vector machines and ensembles like random forests are considered as black boxes (Johansson *et al.*, 2004; Olson *et al.*, 2012; Kamwa *et al.*, 2012b).

Currently there is no common ground on the subject of tree and rule model complexity. Although considered as transparent, critics note that the interpretative value of complex tree and rule models should be questioned (Johansson *et al.*, 2004). On the other hand, a study on model understandability found indications that the assumption where simpler models are considered as more understandable does not always hold as true either (Allahyari *et al.*, 2011).

The choice between a transparent and non-transparent modelling technique is not immediately obvious since there is a tradeoff to be made between accuracy and transparency. Black box modelling techniques generally have better classification and prediction performance, but the tradeoff with better interpretable solutions is unavoidable. We found two solutions in the body of literature that aim to bridge this gap.

The first solution is aimed towards extracting comprehensible information in the form of rules and trees from black box modelling techniques like artificial neural networks and support vector machines (Johansson *et al.*, 2004; Martens *et al.*, 2007; Setiono, 2003). The practice delivers comprehensible information but is criticized for being unrepresentative of the original model due to oversimplification (Cortez *et al.*, 2013).

The second solution approaches the problem from the opposite direction by improving the performance of a transparent modelling technique to a level where it competes with its black box counterparts. A variant of linear modelling is applied known as generalized additive modelling (GAM) enriched with information on pairwise interactions between features (Lou *et al.*, 2013). This allows to retain the explanatory value of linear models and at the same time achieve high performance in terms of classification accuracy. The technique exposes the contribution of each feature in relation to the outcome values.

6 METHOD FRAGMENTS

In this section we present the method fragments as derived from our literature study on the domains of data mining and machine learning. All analytical recipes are accompanied with a brief description.

6.1 Data Understanding

Before starting with any data mining project it is important to become familiar with the data that will be analyzed. The goal is to improve one's understanding of the data by using (statistical) tools to summarize, plot and review datapoints in the data set. This practice is called exploratory data analysis (EDA) (Tukey 1977).

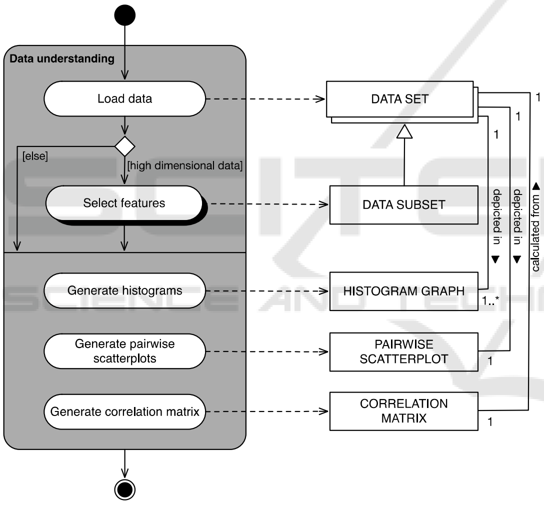

Figure 2: Data understanding method fragment.

The data understanding phase as depicted in Figure 2 revolves around the application of exploratory data analysis (EDA) techniques to generate visualizations and tables to gain a first insight into the relationships between the features of a data set. A high number of features can make these deliverables difficult to interpret. Therefore, the activity flow shows that in cases of high dimensional data sets it is recommended to pre-select a subset of features using a feature selection technique.

We recommend the creation of histogram graphs, pairwise scatterplots and correlation matrices to start exploring relationships between the features of a dataset. Histogram graphs and pairwise scatterplots serve the purpose of visualizing overlap and separability between the various classes of a data set. Feature correlation matrices are used to determine which features are redundant; these should be removed when applying the naive bayes (probabilistic) model. Menger *et al.* (2016) notably provide a more detailed recipe for performing interactive visualisation-driven EDA.

6.2 Data Preparation

The data preparation phase (Figure 3) consists of three main activities: dataset construction, feature extraction, and modelling technique preparation.

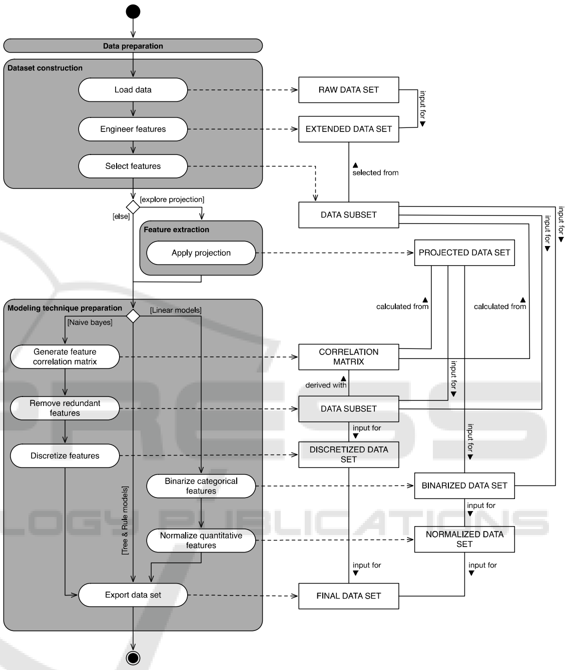

Figure 3: Data preparation method fragment.

Dataset construction: The dataset construction activity entails loading the raw data and engineering new features based on the raw data. Feature engineering can be a substantial task but is difficult to capture in a method since it is highly situational. The last task within this activity is feature selection. Not all features in a given data set have the same informative importance or any importance at all. This can be problematic as some classification algorithms are designed to make the most of the data that is presented to them. In these cases even irrelevant features will eventually be included in the model. In other words the model will be overfitted to the data which means that the classification algorithm has included the noise as an integral part of the model (Tang, Alelyani, and Liu, 2014). The solution is to select a subset of only the most informative features reducing the dimensionality (number of features) of the data set in the process. Feature selection is either performed manually using EDA techniques, or selection is performed using a feature selection algorithm.

Feature extraction: The feature extraction activity entails the application of projection methods. Projection methods like principal component analysis are automated feature engineering techniques that aim to best describe the main differentiators of a data set creating a select (low) number of features in the process (dimensionality reduction). Transparency between the outcome variable and the original features may be lost while using a projection technique.

Modelling technique preparation: Lastly, the modelling technique preparation activity consists of three paths that define preparation steps depending on the model type chosen by the data scientist. When tree and rule models are required due to model transparency concerns, no additional preparation steps are necessary since modern algorithm implementations take care of preparation steps internally. Linear models and the probabilistic naive Bayes model can be chosen due to performance concerns. Both types require their own conversion steps in order to be able to process the data in the next phase of the DM process. The naive Bayes model type e.g. requires redundant features to be removed since they will negatively influence classifier results. Linear model types require input data to be represented in numerical form so transformation steps should be performed as needed e.g. the binarization of categorical data. Note however that some concrete algorithm implementations of linear models may perform these steps as part of their internal workings.

6.3 Modelling and Evaluation

The modelling and evaluation method fragment (Figure 4) consists of three activities aimed at deriving classification models from data sets: search space definition, find optimal parameters, and predict & classify.

Search space definition: The search space definition activity has a route to explore fully automated model (and parameter) selection in analyzing the data set. Currently one experimental implementation exists in the form of Auto-WEKA (Thornton *et al.* 2013). Auto-WEKA is an experimental machine learning toolkit that almost completely relies on Bayesian optimization techniques to generate models. The toolkit is unique

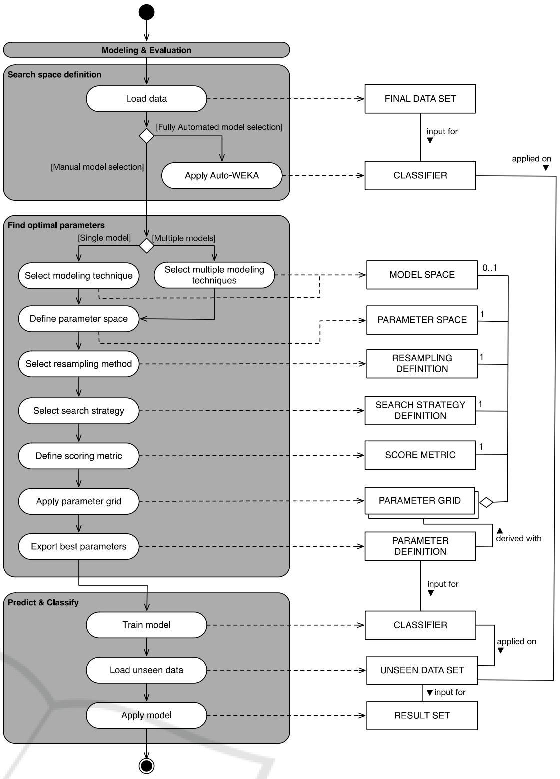

Figure 4: Modelling & evaluation method fragment.

in the sense that it considers the choice for the modelling technique as part of the problem space as well. This relieves potential users from having to manually select and test algorithms, instead Auto-WEKA uses all the algorithms that are part of the WEKA toolkit and determines which algorithm generates the best results for a given data set. Currently, due to the novelty of this technique, the approach should be used to gain initial insight into model types that may perform best on the provided data set.

Find optimal parameters: Next, the application of automated search strategies is central to the following activity named "Find optimal parameters". Recall from our problem statement that the performance of algorithms is highly dependent on how they are configured, a problem known as (hyper) parameter optimization. Getting optimal performance from a modelling technique means finding the right (combination of) parameter settings. The best settings will be different for each data set which necessitates an automated means of determining these values. Search strategies like grid search, random search and Bayesian optimization support the task to (intelligently) iterate over combinations of parameters evaluating the performance at each attempt.

This task requires the data scientist to decide on various factors that determine how the search for the best configuration will be executed. We recommend to consider at least the following "Top 5" factors:

1. *Model type:* The model type itself. The data scientist can choose to iterate over different modelling techniques (tree, rule, ensemble, linear and probabilistic) to find out which type works best given a specific data set. This approach is similar to Auto-WEKA since it includes the model type as part of the problem (search) space.
2. *Parameter types:* This key factor comprises the parameters that belong to a specific model type. Parameter types can range from procedural configuration settings to the specific number of times a procedure is performed.
3. *Resampling method:* The resampling method used to support the evaluation process. Resampling methods apply various procedures to train and test models on the data provided to them. For example, the holdout method splits the data set in a training and test set, usually in a 70%-30% ratio. The model is first trained using the training set, afterwards it is tested on the unseen instances of the test set. Other resampling methods include: (stratified) k-fold cross validation, leave-one-out and bootstrapping.
4. *Search strategy:* The search strategy itself. Grid search is exhaustive by nature, meaning that all possible parameter combinations will be tried. This can be costly both in time and computing resources. Random search and Bayesian optimization aim to find the optimal set of parameters intelligently requiring significantly less tries to do so.
5. *Performance metrics:* The performance measure(s) used to evaluate each attempt. Common measures are classification accuracy, true positive rate (TPR), false positive rate (FPR) and the area under the curve (AUC). Using a combination of measures is necessary since classification accuracy by itself is known to misrepresent the performance of a model in the case of class imbalances in the data set.

The factors discussed above are common to the search strategies outlined in this section, and combined they form the template that makes up the complete problem space through where the search will be executed. The structure and accessibility of this approach is in line with the design goal of this research where we aim to construct a method that enables a user to create optimal models.

Predict & classify: Lastly, the activity "predict & classify" is followed to conclude a DM project. The model derived from the parameter search activity can now be used to classify new and unseen data.

7 FUTURE RESEARCH

We are currently extending and refining the method fragments as outlined in Figures 2-4 with the goal to ultimately evaluate the method on a broad array of data sets, ranging from small/large to low/high dimensional data sets. We are interested to see how classification performance holds up over different variants in data sets. We are also especially interested, by using qualitative research methods, in studying to what extent the methods support non-data scientists in their efforts to perform DM projects.

Next, the problem space of our research could be broadened to cover cases outside of the domain of supervised binary classification, e.g. multiclass, regression and image analysis problems. Method fragments could be created to deal with (sub)cases in the aforementioned domains.

Furthermore, the structures defined in these methods could be used for the development and enhancement of data mining tools. Auto-WEKA is an example of such a tool but follows a rigid method. For example, the tool uses a pre-set path of actions and tasks and does not support embedding domain knowledge during the DM process. From our own experiences we identify a great need for sophisticated tools that offer simplified access to advanced ML techniques while retaining the ability to embed domain knowledge in the data mining process.

Finally, we aim to further refine and integrate existing meta-algorithmic models, as well as to incrementally yet continuously broaden our modelling scope in creating ML method fragments to also include unsupervised learning, non-binary classification tasks, and unstructured data (*e.g.* Spruit and Vlug, 2015), among others.

As our strategic objective we envision Meta-Algorithmic Modelling (MAM) as a well-defined, transparant, and methodological infrastructure for Applied Data Science (ADS) research which has the potential to uniformly interconnect the vast body of knowledge as recipes for machine learning by enabling application domain experts to reliably perform data science tasks themselves in their daily practices.

REFERENCES

Allahyari, H., and N. Lavesson. 2011. “User-Oriented Assessment of Classification Model Understandability,” in *11th Scandinavian Conference on Artifical Intelligence,* pp. 11-19.

Brinkkemper, S. 1996. “Method Engineering: Engineering of Information Systems Development Methods and Tools,” *Information and Software Technology (38:4)*, pp. 275-280.

Cortez, P., and M. J. Embrechts. 2013. “Using Sensitivity Analysis and Visualization Techniques to Open Black Box Data Mining Models,” *Information Sciences (225)*, pp. 1-17.

Domingos, P. 2012. “A Few Useful Things to Know about Machine Learning,” *Communications of the ACM (55:10)*, pp. 78-87.

Hall, M., E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. 2009. “The WEKA Data Mining Software: An Update,” *ACM SIGKDD Explorations Newsletter (11:1)*, pp. 10-18.

Hevner, A., S. March, P. Jinsoo, and S. Ram. 2004. “Design Science in Information Systems Research,” *MIS Quarterly (28:1)*, pp. 75-105.

Johansson, U., L. Niklasson, and R. König. 2004. “Accuracy Vs. Comprehensibility in Data Mining Models,” in *Proceedings of the Seventh International Conference on Information Fusion Vol. 1,* pp. 295-300.

Kamwa, I., S. Samantaray, and G. Joós. 2012. “On the Accuracy Versus Transparency Trade-Off of Data-Mining Models for Fast-Response PMU-Based Catastrophe Predictors,” *IEEE Transactions on Smart Grid (3:1)*, pp. 152-161.

Kotsiantis, S. B., I. Zaharakis, and P. Pintelas. 2007. “Supervised Machine Learning: A Review of Classification Techniques,” in *Emerging Artifical Intelligence Applications in Computer Engineering, pp. 3-24.*

Lou, Y., R. Caruana, J. Gehrke, and G. Hooker. 2013. “Accurate Intelligible Models with Pairwise Interactions,” in *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,* pp. 623-631.

Menger, V., M. Spruit, K. Hagoort, and F. Scheepers. 2016. “Transitioning to a Data Driven Mental Health Practice: Collaborative Expert Sessions for Knowledge and Hypothesis Finding,” *Computational and Mathematical Methods in Medicine*, Article ID 9089321.

Olson, D. L., D. Delen, and Y. Meng. 2012. “Comparative Analysis of Data Mining Methods for Bankruptcy Prediction,” *Decision Support Systems (52:2)*, pp. 464-473.

Orloff, M. 2016. “*ABC-TRIZ: Introduction to Creative Design Thinking with Modern TRIZ Modelling*,” Springer.

Pachidi, S., M. Spruit, and I. van der Weerd. 2014. “Understanding Users' Behavior with Software Operation Data Mining,” *Computers in Human Behavior (30)*, pp. 583-594.

Pachidi, S., and M. Spruit. 2015. “The Performance Mining method: Extracting performance knowledge from software operation data”, *International Journal of Business Intelligence Research (6:1)*, pp. 11–29.

Pritzker, P., and W. May. 2015. *NIST Big Data interoperability Framework (NBDIF): Volume 1: Definitions*. NIST Special Publication 1500-1. Final Version 1, September 2015.

Setiono, R. 2003. “Techniques for Extracting Classification and Regression Rules from Artificial Neural Networks,” *Computational Intelligence: The Experts Speak Piscataway, NJ, USA: IEEE*, pp. 99-114.

Simke, S. 2013. “*Meta-Algorithmics: Patterns for Robust, Low Cost, High Quality Systems*,” Wiley – IEEE.

Spruit, M., and B. Vlug. 2015. “Effective and Efficient Classification of Topically-Enriched Domain-Specific Text Snippets”, *International Journal of Strategic Decision Sciences (6:3)*, pp. 1–17.

Tang, J., S. Alelyani, and H. Liu. 2014. “Feature Selection for Classification: A Review,” *Data Classification: Algorithms and Applications Vol. 37*, pp. 2 – 29.

Thornton, C., F. Hutter, H. H. Hoos, and K. Leyton-Brown. 2013. “Auto-WEKA: Combined Selection and Hyperparameter Optimization of Classification Algorithms,” in *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 847-855.

Tukey, J. W. 1977. “*Exploratory Data Analysis*,” Addison-Wesley.

van de Weerd, I., and S. Brinkkemper. 2008. “Meta-Modelling for Situational Analysis and Design Methods,” *Handbook of Research on Modern Systems Analysis and Design Technologies and Applications*, pp. 35-54.

Vleugel, A., M. Spruit, and A. van Daal. 2010. “Historical data analysis through data mining from an outsourcing perspective: the three-phases method,” *International Journal of Business Intelligence Research, (1:3)*, pp. 42-65.

Yoo, I., P. Alafaireet, M. Marinov, K. Pena-Hernandez, R. Gopidi, J. Chang, and L. Hua. 2012. “Data Mining in Healthcare and Biomedicine: A Survey of the Literature,” *Journal of Medical Systems (36:4)*, pp. 2431-2448.