Proceedings of the of the 17th Conference on Computer Science and Intelligence Systems pp. 403–412
DOI: 10.15439/2022F296
ISSN 2300-5963 ACSIS, Vol. 30

Considering various aspects of models' quality in the ML pipeline - application in the logistics sector

Eyad Kannout*, Michał Grodzki † and Marek Grzegorowski ‡
Institute of Informatics, University of Warsaw
Banacha 2, Warsaw, Poland
Email: *eyad.kannout@mimuw.edu.pl, †m.grodzki@students.mimuw.edu.pl, ‡m.grzegorowski@mimuw.edu.pl

***Abstract*—The industrial machine learning applications today involve developing and deploying MLOps pipelines to ensure the versatile quality of forecasting models over an extended period, simultaneously assuring the model's accuracy, stability, short training time, and resilience. In this study, we present the ML pipeline conforming to all the abovementioned aspects of models' quality formulated as a constrained multi-objective optimization problem. We also provide the reference implementation on state-of-the-art methods for data preprocessing, feature extraction, dimensionality reduction, feature and instance selection, model fitting, and ensemble blending. The experimental study on the real data set from the logistics industry confirmed the qualities of the proposed approach, as the successful participation in an international data competition did.**

***Index Terms*—XGBoost, Dimensionality reduction, Ensemble blending, Feature selection, Feature extraction, MOO, MCDA, Logistics**

I. Introduction

MACHINE learning (ML) algorithms are widely used within decision-support [1] or recommender systems [2] in many branches of the industry, like fast-moving consumer goods (FMCG) [3], e-commerce [4], logistics [5], or even hard-coal mining [6]. However, as ML models continue to run in production environments for an extended period, new expectations and concerns have also arisen. Some time ago, data scientists were expected to deliver a fine-tuned model, today, attention is paid to building ML operations (MLOps) pipelines responsible for continuous monitoring and ensuring the quality of the developed models during their functioning [7], [8]. It is also worth paying attention to the ongoing shift in the quality assurance of models' predictions, which are no longer limited to optimizing a single measure, such as accuracy or root mean square error (RMSE). It should cover more models' characteristics like stability [9], resilience [10], or interpretability [11].

The fundamental task of data analysis is to represent the data accordingly to the investigated problem. The selection of appropriate criteria for assessing the quality of generated predictions is no less important. The choice of such measure primarily depends on the nature of the problem, e.g., there are different ones for classification and regression. The quality measure we choose in the model optimization process will have a crucial impact on its performance. Once decided, we can rely on AutoML meta-learning to ignite a versatile exploration of several learning algorithms meanwhile optimizing their parameters, which, however, is very costly and time-consuming [12]. Whereas, over-optimizing a single measure in many applications is simply unnecessary. In particular, further tuning a model of sufficient quality may lead to over-fitting, increase complexity, and reduce interpretability, not to mention the longer learning time and the increased cost of computing resources. Furthermore, in many cases, optimizing a single quality measure is insufficient. Reaching the optimal regression model according to the RMSE, which meanwhile is vulnerable to data deficiencies (e.g., unavailability of selected attributes), is pointless. One of the ways we may address those concerns is to refer to the multi-objective optimization (MOO) [13]. However, as the result of MOO, we do not end up with a single solution but many Pareto optimal models. Selecting the best one is still a complex and time-consuming task related to the multi-criteria decision aiding (MCDA) [14].

In response to the above expectations and challenges, let us present the ML pipeline for training forecasting models that allows optimizing not only a single quality measure, such as RMSE, accuracy, or F1-score, but also taking into account the robustness and resilience of the ensemble blended. The developed pipeline assumes that during the training procedure, the goal is not to optimize the model over days to achieve even a minimal quality improvement on a single error measure but to adhere to many business expectations possibly fast. Accordingly, we define the task as a MOO and adapt ε-constrained scalarization for the investigated criteria [14]. By referring to quality thresholds that correspond directly to business expectations, we could significantly limit the time of model fitting (from days to seconds), which obviously determines the lower cost of cloud computing resources [15]. Furthermore, the adopted principle of building a model on random subsets of attributes and rows allows to achieve a variety of different models within an ensemble [16], [17]. Such an approach to training set selection enables a very straightforward parallelization of the learning procedure and hence provides significant acceleration of computation [18].

Yet another material aspect of the developed solution is the proposed feature extraction mechanism. The method is composed of several steps. Firstly, we combine available data sources into one flat file and aggregate the one-to-many relations with the common $SELECT \ldots GROUP\ BY$ SQL-

Research co-funded by Polish National Science Centre (NCN) grant no. 2018/31/N/ST6/00610.

based approach to extract some generic statistics. Later, we use feature extraction methods like one-hot encoding and ordinal coding to obtain a numerical representation of the data. This way, we achieve a sparse data representation, which poses a big problem for the boosting tree algorithms by impacting the quality of their cuts on the attributes. Such a situation imposes the construction of deeper trees, making generalization difficult and leading to over-fitting. Therefore, after encoding a given feature, we apply one of the most popular dimensionality reduction methods - principal component analysis (PCA) - to use the first few components.

To show the particular qualities of our solution, we present a case study in the logistics industry for predicting costs associated with forwarding contracts. For this purpose, we used three data sets from the machine learning contest organized on the KnowledgePit.ml platform [19], which we combined together, preprocessed, and analyzed with the developed solution. In the conducted research, we assume that the acceptable level of the prediction error measured with the RMSE measure should not exceed 2.5% of the average cost of forwarding contracts in data that corresponds to RMSE of approx 0.17. We also assume the robustness threshold of 0.02, understood as the maximal acceptable difference of RMSE achieved by the model on the training and validation set during the training procedure. Furthermore, we set the resilience threshold so the constructed ensemble should consist of at least 10 models. This way, we could provide reliable forecasts even if some of the models within the ensemble became unreliable and could impair the overall prediction quality of the ensemble. Such a situation may occur in production environments, e.g., due to a software error or unavailability of the critical attributes for this model.

The main contributions of this paper are as follows:

1) The ML pipeline considering various aspects of models' quality formulated as a constrained multi-objective optimization problem.
2) The complete reference implementation of the ML pipeline providing methods for preprocessing, feature extraction, dimensionality reduction, feature and instance selection, model fitting, and ensemble blending.
3) The experimental study on the real data set from the logistics industry that confirmed several qualities of the proposed approach, including small prediction error (RMSE), robustness to over-fitting, fast computing time, and resilience.

The rest of the paper is organized as follows. In Section II, we review the related literature. Section III provides a complete reference for the developed ML pipeline. In Section IV, we describe in detail the experiments conducted in this study including the description of the data, experimental setup, and the results. Finally, in Section V, we draw conclusions and suggest possible future research directions.

II. Related Works

Due to the general availability and affordability of cloud services [15], and the proven effectiveness of machine learning [5], [17], modern enterprises massively automate their processes and optimize decision-making with intelligent use of the collected data [3]. This trend is beneficial to many industries, including supply management and logistics [20]. Let us pay special attention to international freight transportation, which is related to moving goods between countries and may involve many stakeholders: shippers, carriers, forwarders, third-party logistics services, and customs of two or more countries for each movement [21]. In this context, machine learning is seen as one of the primary enablers for the dynamic development of enterprises, allowing for apt data-driven decisions, including route planning, travel time prediction, vehicle scheduling, estimated time of arrival, and foremost accurately predicting costs related to the execution of forwarding contracts [22], [23].

We can model this task as a regression of the forwarding contract costs conditioned by the attributes of orders, such as the type of order, basic characteristics of the shipped goods (e.g., dimensions, special requirements), and the expected route that a driver will have to cover. Among the ML algorithms commonly applied to solve the regression problems, we may point out eXtreme gradient boosting trees (XGBoost), deep neural networks, or support vector machines [21], [24]. Considering the industry specifics and the dynamics of changes in the business and technological environment, the developed data-driven decision-making system should promptly adapt to changes and operate reliably even in the event of data deficiencies. One of the ways to simultaneously address several potentially conflicting concerns is multi-objective optimization (MOO) [13].

Classically, MOO problems are often solved using scalarization techniques. In brief, scalarization means that the objective functions are aggregated (or reformulated as constraints), and then a single-objective problem is solved [25]. However, this method requires defining the perfect balance between objectives' importance. Another possibility to solve such a problem is to rely on Pareto front (PF) methods. For instance, the ϵ-constraint approach can obtain a set of PF solutions by keeping only one objective and subdividing the others into several segments with some thresholds. Here, we do not end up with a single solution but potentially many models, and selecting the optimal one requires further effort [14]. In the proposed framework, we refer to ϵ-constraint filtering, but instead of choosing a single model, we blend the ensemble of several solutions [17]. This way, we not only avoid the multi-criteria decision task but also introduce the additional resilience level to our solution [10].

Among the popular ensembling techniques, we may mention random forest and XGBoost. These approaches of blending tree models minimize the regression (or decision) trees' tendency of overfitting, hence, ensuring better robustness and stability [9], [24]. The stability, RMSE, and resilience can be further improved by ensuring that the trained models in the final ensemble are relatively different from each other. One way to do this is to train models on diverse subsets of objects and attributes [17]. The training set selection, complemented by parallelization of computation, can lead to better general-

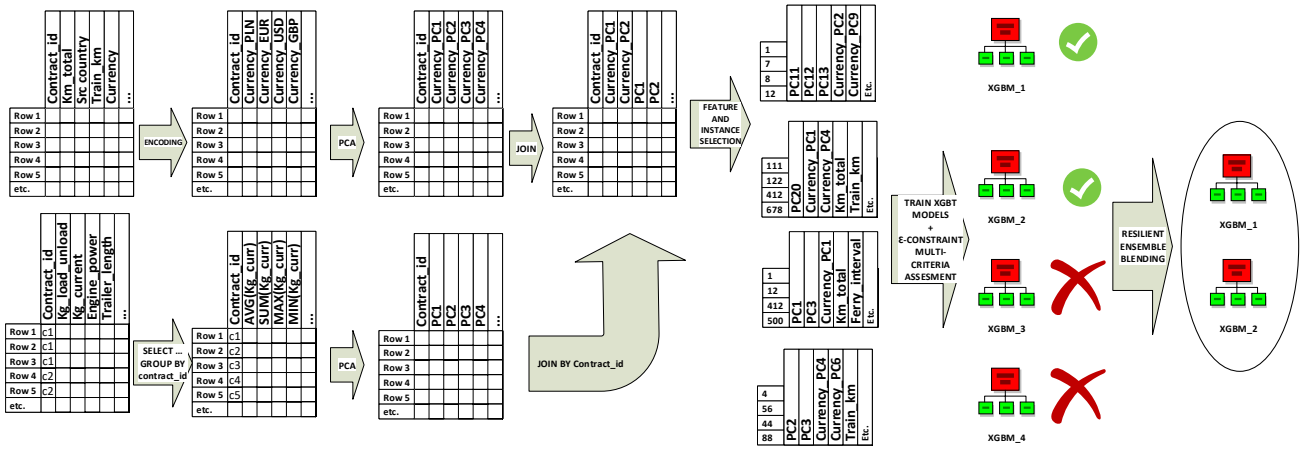

Fig. 1: Schematic ML pipeline implementation for the logistics data.

ization and minimize the overall training latency [16], [18]. It is usually essential to ingest several diverse data sources to provide adequately rich data representation with many different attributes. However, models such as XGBoost or deep neural networks operate on numerical features, whereas in most databases, we will also find some other data types.

A typical approach, in this case, would be to apply feature extraction, in particular, to encode each feature in numerical form [26]. One of the weaknesses of this approach may be the creation of a very sparse data representation related to the one-hot encoding of categorical features, which in turn can have a negative impact on tree models' performance.

Furthermore, as the number of features increases, the model training takes far more time and consumes more resources. There are many methods allowing to project or embed the data into a lower-dimensional space while retaining as much information as possible, just to mention independent component analysis, multidimensional scaling, or principal component analysis (PCA) [26]. The central idea of PCA is to reduce the dimensionality retaining as much as possible of the variation present in the data [27]. In the proposed framework, we refer to all the above-mentioned techniques. Furthermore, putting the things together, we propose the end-to-end ML pipeline adhering to the ML operations paradigm that ensures the solution's quality over time.

III. Solution Overview

The developed ML pipeline consisted of several stages related to data ingestion, preprocessing, extending representation, reducing dimensionality, robust model training, and resilient ensemble blending. The first step in the developed pipeline is data ingestion and integration. Next, we perform data cleansing, encoding categorical variables to the numeric form, extracting some custom characteristics from text columns, imputing missing values, and conducting further feature extraction (FE) [18], [28].

FE addresses the problem of finding the most compact and informative data representation and is fundamental for every ML pipeline. The importance of proper data representation was aptly identified by *Pedro Domingos: "At the end of the day, some machine learning projects succeed, and some fail. What makes the difference? Easily the most important factor is the features used"*. Using more relevant data sources and better knowledge representation may have a crucial impact on the final model quality. Therefore we plan to further extend the developed FE methods by introducing histogram-based feature engineering [29]. Among other relevant, recently reported approaches that could be in the future implemented in our ML framework, we may indicate embedding selected statistics from survival analysis or features derived from deep learning methods into data representation [5], [30].

These activities sometimes require additional effort that may depend on the data. Hence, they may not always be fully automated. For example, in the discussed case study of forwarding contracts, we ingest three data sources: css_main, css_routes, and fuel_prices tables (cf. Section IV-A). However, to integrate css_routes, we have to aggregate the data first. We execute this with the aggregating query: *SELECT AVG(.), MIN(.), MAX(.), SUM(.), COUNT(.) ... GROUP BY id_contract* for each interesting variable in the table. Considering that the attributes obtained in this way are not intended for financial settlements but to feed the machine learning procedure, a vital extension of our approach would be to relay on approximated results of SQL instead of exact ones [31]. Approximate query engines can generate summaries of Big Data sets much faster with only a slight loss in precision, which may be negligible in model generalization [32], [33].

Instead of using all the SQL results for fitting the predictive model, the outcomes of the aggregation queries are transformed with PCA, and several first components are integrated into the main data. The schematic view of this process, related to the discussed case study, you may find on the left part of Figure 1. In general, whenever the encoding of categorical features into numerics significantly increases data sparsity, the derived variables are encoded with PCA, and a few first

components are kept. At the same time, the rest may be omitted. To provide an example, in Figure 2, we present the variance explained by the first 10 principal components (PC) of one-hot encoded first_load_country attribute.

The central part of the pipeline is selecting the training set, i.e., features and instances, to achieve the best performance and robustness of the models. In the developed approach, we iteratively draw a subset of attributes and instances as a ratio of the original data controlled by two thresholds: ω_r and ω_c (cf. Algorithm 1). In the next step, we train the selected predictive model (e.g., XGBoost or LightGBM) [34]. Since we fit the predictive model to significantly smaller data chunks, we naturally minimize the time of this process. The selection of training subsets is random and independent from each other. Hence, this process may also be easily parallelized, e.g., by drawing several subset candidates simultaneously and training the models in parallel. We also see a potential to extend our framework with the heuristic search over the subsets of attributes and features to reach the optimal quality (i.e., minimalize reported error) faster [35]–[37], and to introduce more advanced feature selection techniques [26]. In this context, granular feature selection techniques could be a perfect fit [38], including r-C-reducts, bi-reducts [39], or reviving the concept of dynamic reducts [40]. Besides more advanced feature selection algorithms, instance selection has space for improvement as well. Ordering the records by date and considering only the newest instances while drawing the subsets for training data is one of many possible ideas. Combining those in an ensemble could yield interesting results.

Data:
- *dataTrain*, *dataTest* - training and test data
- θ - acceptable RMSE threshold
- ϑ - stability threshold for RMSE
- ρ - expected resilience level
- ω_r and ω_c - instance and feature selection thresholds
- *score()*- quality measure, e.g., RMSE
- N - maximal number of unsuccesful attempts

Result: *ensemble of models*

```
/* Initialization                                   */
ensemble ← ∅; k ← 0
validationSet ← dataTrain.sample
dataTrain ← dataTrain \ validationSet
while |ensemble| < ρ ∧ k < N do
    trainSet ← draw ω_r rows and ω_c cols from dataTrain
    model ← trainXGBT(trainSet)
    Θ_t ← score(model, trainSet)
    Θ_v ← score(model, validationSet)
    if Θ_t < θ ∧ Θ_v < θ ∧ |Θ_t − Θ_v| < ϑ then
        k ← 0
        ensemble ← ensemble ∪ {model}
    else
        k ← k + 1
    end
end
return ensemble;
```

Algorithm 1: Resilient and stable ensemble blending

Each model fitted on training subsets is assessed with the ε-constrained approach to handle multiple quality criteria, as

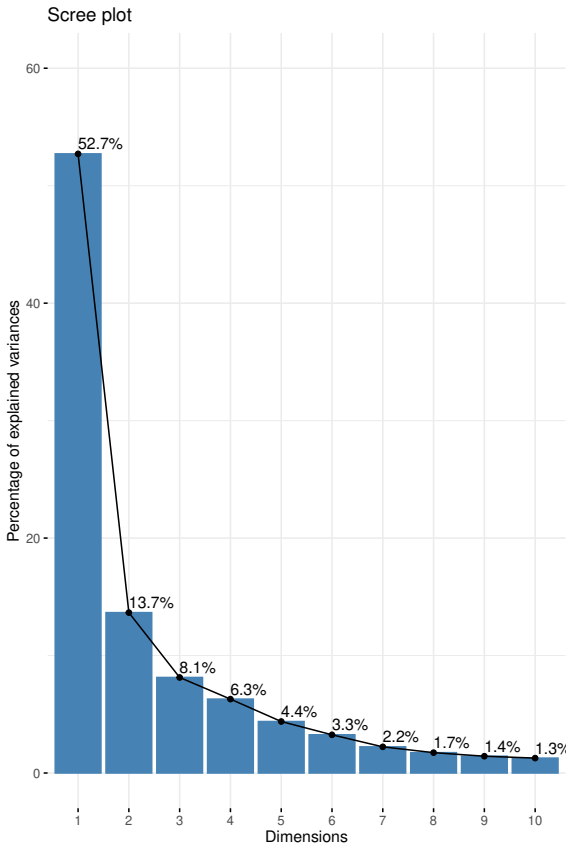

Fig. 2: Variance explained by the first ten PCA components of one-hot encoded first_load_country attribute.

presented in Algorithm 1. In particular, the primary objective function is to minimize an error measure, e.g., RMSE. We guarantee that each of the trained models yields an error lower than the predefined threshold θ on validation data. Furthermore, we introduce a stability threshold ϑ to avoid overfitting. We calculate it as an absolute difference between errors reported on training and validation sets in each round. The best models are blended to form an ensemble of possibly diverse models. The additional parameter ρ determines the number of models within the ensemble to provide a certain resilience level in case some o the models were put out of action.

In the future, we plan to extend a multicriteria evaluation with a specific approach to assure difference between the models explicitly. One of the possible solutions could be measuring the distances between the reported scores on a validation set. Alternatively, we may assure the feature importance rankings reported by models are possibly dissimilar (cf. Figures 4). Another approach to construct ensembles of possibly diverse models could be achieved by referring to r-C-reducts on nonoverlapping feature subsets or by ensuring constraints between attribute sets [10], [41]. A combination of the above techniques would also be an exciting future research direction, primarily since many attributes are somehow related,

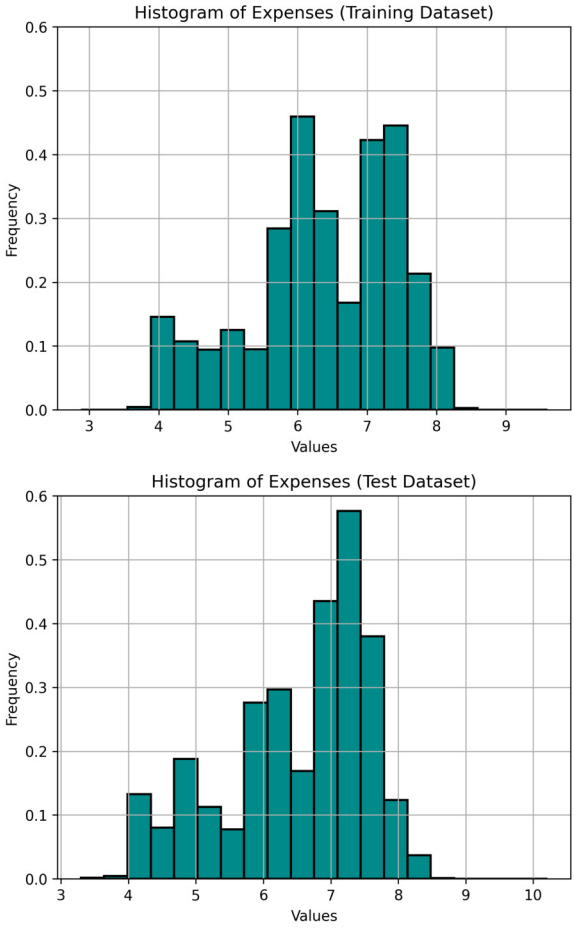

Fig. 3: Histogram of the target variable (expenses) in the experimental data (on top) and the FedCSIS competition's final dataset (bottom).

e.g., principal components generated from the same original features or all the columns in databases referring to the same source of knowledge. Combining these techniques with state-of-the-art instance selection and training set selection methods would also be of great importance [42].

Both in wrapper search and model training, we focused mainly on XGBoost [43]. However, we also complemented it with an evaluation of LightGBM [34]. We conducted grid search model parameter tuning for best performing data subsets in the feature selection stage This part was conducted iteratively and was alternated with the wrapper-based feature selection. Such an iterative approach allowed us to, over time, fine-tune the wrappers' configuration. In the future, we plan to continue experiments with more advanced techniques of searching the hyper-parameter space in order to optimize the model faster and more efficiently [44].

IV. Experiments

A. Data

The data sets contain 6 years of history of orders appearing on the transport exchange, along with details such as the type of order, basic characteristics of the shipped goods (e.g., dimensions, special requirements), as well as the expected route that a driver will have to cover (cf. Table I). In particular, the training data consist of two tables: css_main_training.csv and css_routes_training.csv, and the additional data table containing historical wholesale fuel prices for the period of training and test data. The first file (i.e., css_main_training.csv) contains basic information about the contracts, and the second one (i.e., css_routes_training.csv) describes the main sections of the planned routes associated with each contract. In both tables, the first column (i.e., id_contract) contains identifiers that allow matching records from css_main_training.csv and css_routes_training.csv files. Additionally, the second column in the css_main_training.csv file (i.e., expenses) contains information about the prediction target. Values in this column are available only for the training data.

B. Task and experimental setup

In our study, we investigated the task of predicting the costs related to the execution of forwarding contracts, which was defined within the 8th data mining competition organized online on the KnowledgePit platform in association with the Federated Conference on Computer Science and Information Systems (FedCSIS'22). The task is to design an accurate method for regression of costs associated with forwarding contracts [45], based on contract data and planned routes (cf. Section IV-A). The quality of predictions was evaluated with the RMSE measure. The experiments were also planned to validate the relation between training speed and the size of training data, controlled with the ω_r and ω_c parameters.

Many threats may impact the models' performance or impede their operations, including missing data or software errors. Therefore, the experimental setup was designed to also evaluate the resilience of the final solution, understood as the RMSE error achieved when some models within the ensemble cannot be applied (e.g., due to a software error or missing data attributes). However, up to our knowledge, there is no established methodology allowing us to assess the resilience of the ML models. One of the approaches could be to randomly delete subsets of test data, e.g., by dropping particular columns. It is, however, not straightforward how to implement this kind of test. Shall we randomly drop a certain percent (e.g., 5% or 10%) or number (3 or 5) of all columns? For data sets containing 20000 attributes, such operation may have minimal impact on predictive models [6], [17]. Or shall we drop the model's most important feature(s)? Different approaches would be preferable for other modeling techniques. Consider a random forest that relies on many redundant weak tree models, XGBoost where particular predictors are boosting the formerly selected ones, or a single tree that depends on just a few attributes with one surrogate or verifying cut per split

TABLE I: Competition data description

Data type	Example columns	Description	Processing	
Categorical data	id_payer, id_currency, direction, load_size_type, service_type, contract type, first_load_country, last_unload_country	Information about transport type, start and destination country, contract currency	one-hot encoding and PCA	
GPS data	first_load_lat, first_load_lon, last_unload_lat, last_unload_lon, route_start_lat, route_start_lon, route_end_lat, route_end_lon	GPS coordinates of transport start and end points	NA	
Numerical data	km_total, km_empty, km_nonempty, prim_ferry_line, ferry_duration, ferry_intervals, max_weight	Information about total distances to be covered with each mode of transport, weight of the payload, current fuel prices, aggregated information about the planned route	Aggregation	
Binary data	refrigerated, if_empty	Additional information about the payload, for example if it was refrigerated	NA	
Date data	route_start_datetime, route_end_datetime	When the service was executed	NA	

[46]. In our case, we decided to implement a straightforward approach that may be dedicated to the ensembling techniques. Namely, we were dropping randomly selected models from the ensemble to measure the impact of such an operation on the RMSE.

In the conducted experiments, we used the features extraction, dimensionality reduction, model training, and resilient ensemble blending method, as described in Section III. As the base model, we used XGBoost a [24]. We optimized the model parameters only once on the selected subset of data with the grid search procedure, and since it was not the major point of our research, we kept those parameters later unchanged. The ω_r and ω_c parameters were set to 0.5, meaning that each of the xgbt models within the ensemble was trained on a random subset of 50% features and 50% instances from the training set. The expected model quality threshold θ was set to 0.17, which corresponds to 2.5% of the median expenses in data (cf. Figures 3). The robustness threshold ϑ was 0.02, and the resilience threshold ρ was 10. To visualize the results, we use box plots - a standard way of displaying data distribution by encoding their five key characteristics: minimum, first quartile (Q1), median, third quartile (Q3), and maximum. For the purpose of evaluation, we split the data into three sets: training, validation, and test. The first two constituted a part of the training procedure. The last was used only for evaluation. Furthermore, we present the results achieved by our method within the "FedCSIS 2022 Challenge: Predicting the Costs of Forwarding Contracts" on the preliminary and final competition data.

C. Results

Figure 5 shows how the values of RMSE are spread out for ensemble models on the training, validation, and test data. The results show that RMSE for training, validation, and test dataset are very close to each other avoiding model overfitting. This confirmed that the ensemble yields not only satisfactory quality but also guarantees high robustness and stability, which is especially visible between validation and test sets (cf. Figure 5). This confirms the effectiveness of the multicriterial evaluation, such as acceptable RMSE threshold θ and stability threshold for RMSE ϑ, which are applied while selecting the ensemble models. When we compare the results achieved by our method within our experimental setup with the results achieved during the FedCSIS 2022 Challenge: Predicting the Costs of Forwarding Contracts, we may notice it yielded very similar results on both preliminary and final data sets, 0.165 and 0.161, respectively. The results confirm the predictability, repeatability, and stability of the proposed method.

Next, we evaluate the resilience of our solution. In this experiment, we randomly deleted 1, 3, 5, 7, and 9 models from the ensemble to simulate the scenario when some models would be unavailable. Then, we calculated RMSE against the test dataset. We repeated the procedure 10 times and plotted the results in Figure 6. The results show that RMSE is slightly decreasing when deleting some models from the ensemble. However, the largest impact was spotted when deleting the majority of the models (9 out of 10). In fact, this leads us to investigate the most important features which are very correlated or have a high impact on the target variable. Therefore, we calculated the F-score based on the Information Gain (IG) measure. It is worth noting that IG in the decision/regression tree-based models is a measure of how much information a feature provides about the target feature.

Figure 4 shows the relative importance of features for two selected XGBoost models from the ensemble. The values on the x-axis show the average gain for the top ten features across all splits where those features were used. Observably, the proposed training set selection procedure allowed us to train several significantly different models that rely on different features, which leveraged ensemble smoothing and enabled the high resilience of the final solution. Considering the importance of features for 10 XGBoost models constituting the final solution (cf. Fig 4), we may notice that only seven features were relatively impactful (i.e., F-score $>$ 100) for more than one model, namely: diff_start_end_days, diff_start_end_weeks, direction_PC1, km_nonempty, km_total, last_unload_country_PC2, last_unload_country_PC5.

For some of the potential threats, like data deficiencies, we may notice that several attributes were derived from the same (or similar) sources of information, e.g., features derived from the start and end dates or principal components of one-hot-encoded last_unload_country. Thus, in such a case,

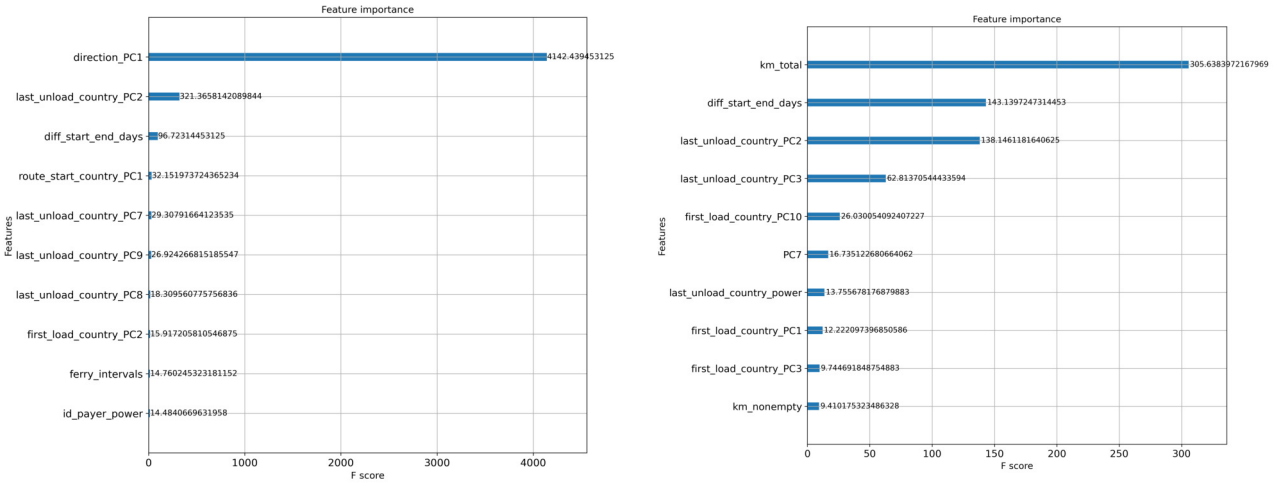

Fig. 4: Feature importance reported by two XGBoost models trained on different subsets of features and instances.

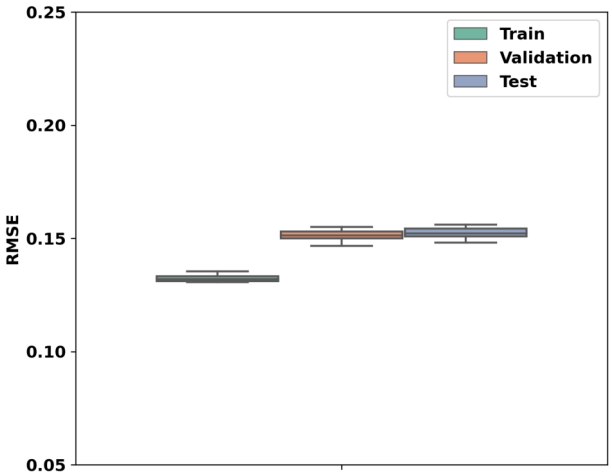

Fig. 5: RMSE comparison for ensemble models.

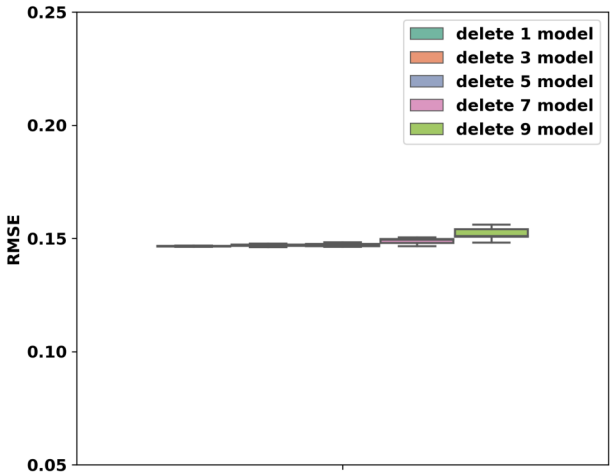

Fig. 6: Resilience comparison for ensemble models.

applying a more formal methodology for resilience assessment would be advisable and may be considered a valid future research direction. Another valuable extension of the proposed framework would be to include information from experts in the machine learning process to ensure that the features which, according to the experts, have high importance are selected in at least some of the ensemble components [26], [47].

Finally, we focus on measuring the speed and cost-effectiveness of the solution. This factor has recently gained a lot of attention because, in many practical applications of ML, like recommender or threat detection systems, the predictions are highly influenced by the most recent data. Thus, the model must be continually retrained to consider the most recent information, and it is very important to make a balance between speed and accuracy. In our proposed solution, this can be achieved by training data sets of randomly selected chunks of data. Figure 7 shows the time taken for building the XGBoost model using 100%, 75%, 50%, and 25% of data rows (cf. ω_r parameter) in the training data set, with the unchanged model hyper-parameters. This experiment was repeated 10 times, considering different (randomly chosen) data rows in each run. Furthermore, in Figure 8, we also plot RMSE each XGBoost model would achieve in FedCSIS'22 data competition. We may notice that depending on the data subset, the final model quality varies and slightly decreases along with the declining size of training data chunks.

V. Summary

The industrial machine learning applications today involve the development and deployment of MLOps pipelines, which consist of automated activities that were once manually performed by data analysts, including data ingestion and preprocessing, feature extraction and selection, model fitting, etc. These solutions are designed to ensure the quality of predictions during the production use of forecasting models, which may be months or even years. Quality assurance in such a long period requires the development of a repeatable

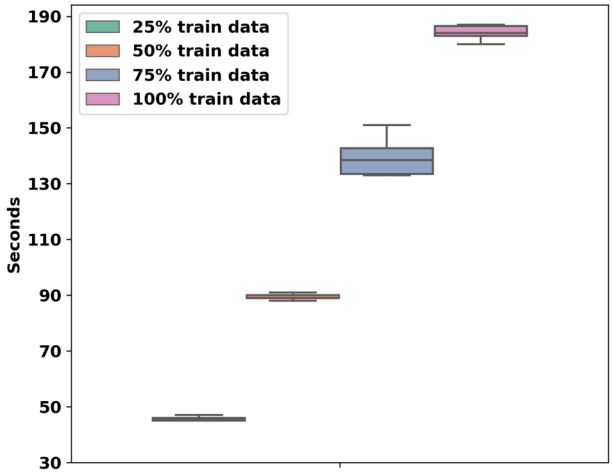

Fig. 7: Training speed comparison.

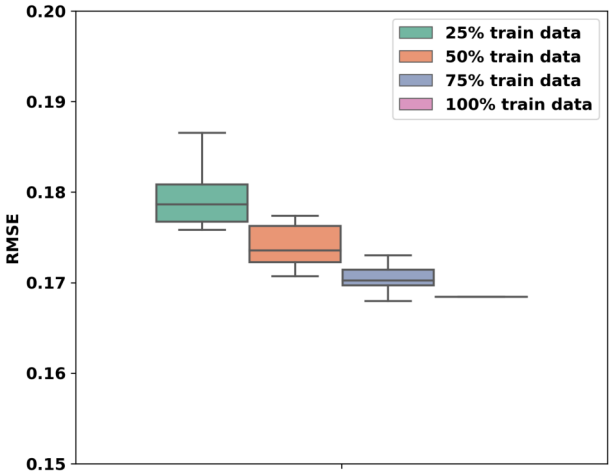

Fig. 8: RMSE comparison before/after deleting data rows.

learning procedure to (re)train the models in case of shifts or drifts in data. Furthermore, apart from confirming the model's accuracy, it is expected to assure many quality criteria, such as stability, resilience, and low computational cost, in a fast and reliable way.

In this study, we present the ML pipeline that considers several qualities of the models in a multicriterial manner. Besides assuring RMSE optimization, our solution also ensures robustness, resilience, and instant (re)training time. The developed pipeline consists of several states, including preprocessing, feature extraction, dimensionality reduction, robust model training, and resilient ensemble blending. In this paper, we also elaborate on several promising future research directions, including applying more advanced features and instances selection techniques, incorporating experts' knowledge into the machine learning processes, ensuring the ensemble diversity more explicitly, or providing a formal methodology to assess the resilience of the predictive models.

We confirmed the qualities of our pipeline with the versatile experimentation on the real data from international freight forwarders and by participating in an international data mining competition organized along to FedSCSIS'22 conference. The achieved RMSE is comparable to the best and most complex models reported by 135 teams from 24 countries in the FedCSIS contest, meanwhile conforming to more requirements. We may conclude that the proposed solution provides high-quality results with excellent resilience and stability, and the models are developed within seconds of training on low-cost compute resources. In the future, we plan to augment the developed framework with the discussed extensions and subject it to in-depth experimental analysis on a more significant number of real data sets from various fields, including the mining industry [17], [18], fire service [28], FMCG [3], cloud resource management [15], and for predicting escalations in customer support [48]. We believe that the developed approach will be equally effective in all those applications.

REFERENCES

[1] E. Zdravevski, P. Lameski, C. Apanowicz, and D. Ślęzak, "From big data to business analytics: The case study of churn prediction," *Appl. Soft Comput.*, vol. 90, p. 106164, 2020. doi: 10.1016/j.asoc.2020.106164

[2] E. Kannout, "Context Clustering-based Recommender Systems," in *2020 15th Conference on Computer Science and Information Systems (FedCSIS)*, 2020. doi: 10.15439/2020F54 pp. 85–91.

[3] M. Grzegorowski, A. Janusz, S. Lazewski, M. Swiechowski, and M. Jankowska, "Prescriptive analytics for optimization of fmcg delivery plans," in *Proceedings of IPMU'22*, 2022.

[4] Y. Li, Y. Yang, K. Zhu, and J. Zhang, "Clothing sale forecasting by a composite gru–prophet model with an attention mechanism," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 12, pp. 8335–8344, 2021. doi: 10.1109/TII.2021.3057922

[5] M. Grzegorowski, J. Litwin, M. Wnuk, M. Pabis, and L. Marcinowski, "Survival-based feature extraction - application in supply management for dispersed vending machines," *IEEE Transactions on Industrial Informatics*, 2022. doi: 10.1109/TII.2022.3178547

[6] D. Ślęzak, M. Grzegorowski, A. Janusz, M. Kozielski, S. H. Nguyen, M. Sikora, S. Stawicki, and L. Wrobel, "A Framework for Learning and Embedding Multi-Sensor Forecasting Models into a Decision Support System: A Case Study of Methane Concentration in Coal Mines," *Information Sciences*, vol. 451-452, pp. 112–133, 2018.

[7] C. Renggli, L. Rimanic, N. M. Gürel, B. Karlas, W. Wu, and C. Zhang, "A Data Quality-Driven View of MLOps," *CoRR*, vol. abs/2102.07750, 2021. [Online]. Available: https://arxiv.org/abs/2102.07750

[8] Y. Zhou, Y. Yu, and B. Ding, "Towards MLOps: A Case Study of ML Pipeline Platform," in *2020 International Conference on Artificial Intelligence and Computer Engineering (ICAICE)*, 2020. doi: 10.1109/ICAICE51518.2020.00102 pp. 494–500.

[9] A. Subbaswamy, R. Adams, and S. Saria, " Evaluating Model Robustness and Stability to Dataset Shift ," in *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, A. Banerjee and K. Fukumizu, Eds., vol. 130. PMLR, 13–15 Apr 2021, pp. 2611–2619. [Online]. Available: https://proceedings.mlr.press/v130/subbaswamy21a.html

[10] M. Grzegorowski and D. Ślęzak, "On resilient feature selection: Computational foundations of r-C-reducts," *Inf. Sci.*, vol. 499, pp. 25–44, 2019. doi: 10.1016/j.ins.2019.05.041

[11] C. Rudin, "Please Stop Explaining Black Box Models for High Stakes Decisions," *CoRR*, vol. abs/1811.10154, 2018.

[12] X. He, K. Zhao, and X. Chu, "Automl: A survey of the state-of-the-art," *Knowl. Based Syst.*, vol. 212, p. 106622, 2021. doi: 10.1016/j.knosys.2020.106622

[13] J. Blank and K. Deb, "Pymoo: Multi-Objective Optimization in Python," *IEEE Access*, vol. 8, pp. 89 497–89 509, 2020. doi: 10.1109/ACCESS.2020.2990567

[14] H. M. Ridha, C. Gomes, H. Hizam, M. Ahmadipour, A. A. Heidari, and H. Chen, "Multi-objective optimization and multi-criteria decision-making methods for optimal design of standalone photovoltaic system: A comprehensive review," *Renewable and Sustainable Energy Reviews*, vol. 135, p. 110202, 2021. doi: 10.1016/j.rser.2020.110202

[15] M. Grzegorowski, E. Zdravevski, A. Janusz, P. Lameski, C. Apanowicz, and D. Ślęzak, "Cost Optimization for Big Data Workloads Based on Dynamic Scheduling and Cluster-Size Tuning," *Big Data Research*, vol. 25, p. 100203, 2021. doi: 10.1016/j.bdr.2021.100203

[16] N. Verbiest, J. Derrac, C. Cornelis, S. García, and F. Herrera, "Evolutionary wrapper approaches for training set selection as preprocessing mechanism for support vector machines: Experimental evaluation and support vector analysis," *Applied Soft Computing*, vol. 38, pp. 10–22, 2016. doi: 10.1016/j.asoc.2015.09.006

[17] A. Janusz, M. Grzegorowski, M. Michalak, Ł. Wróbel, M. Sikora, and D. Ślęzak, "Predicting Seismic Events in Coal Mines Based on Underground Sensor Measurements," *Engineering Applications of Artificial Intelligence*, vol. 64, pp. 83–94, 2017.

[18] M. Grzegorowski, "Massively Parallel Feature Extraction Framework Application in Predicting Dangerous Seismic Events," in *Proceedings of the 2016 Federated Conference on Computer Science and Information Systems, FedCSIS 2016, Gdańsk, Poland, September 11-14, 2016*, ser. Annals of Computer Science and Information Systems, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., vol. 8. IEEE, 2016. doi: 10.15439/2016F90 pp. 225–229.

[19] A. Janusz, D. Ślęzak, S. Stawicki, and M. Rosiak, "Knowledge Pit - A Data Challenge Platform," in *Proceedings of the 24th International Workshop on Concurrency, Specification and Programming, Rzeszow, Poland, September 28-30, 2015*, ser. CEUR Workshop Proceedings, Z. Suraj and L. Czaja, Eds., vol. 1492. CEUR-WS.org, 2015, pp. 191–195.

[20] G. F. Frederico, "From Supply Chain 4.0 to Supply Chain 5.0: Findings from a Systematic Literature Review and Research Directions," *Logistics*, vol. 5, no. 3, 2021. doi: 10.3390/logistics5030049

[21] L. Barua, B. Zou, and Y. Zhou, "Machine learning for international freight transportation management: A comprehensive review," *Research in Transportation Business & Management*, vol. 34, p. 100453, 2020. doi: 10.1016/j.rtbm.2020.100453 Data analytics for international transportation management.

[22] N. Servos, X. Liu, M. Teucke, and M. Freitag, "Travel Time Prediction in a Multimodal Freight Transport Relation Using Machine Learning Algorithms," *Logistics*, vol. 4, no. 1, 2020. doi: 10.3390/logistics4010001

[23] S.-H. Chung, "Applications of smart technologies in logistics and transport: A review," *Transportation Research Part E: Logistics and Transportation Review*, vol. 153, p. 102455, 2021. doi: 10.1016/j.tre.2021.102455

[24] J. Nobre and R. F. Neves, "Combining Principal Component Analysis, Discrete Wavelet Transform and XGBoost to trade in the financial markets," *Expert Systems with Applications*, vol. 125, pp. 181–194, 2019. doi: 10.1016/j.eswa.2019.01.083

[25] R. Kasimbeyli, Z. Kamisli Ozturk, N. Kasimbeyli, G. Dinc Yalcin, and B. İçmen Erdem, "Comparison of Some Scalarization Methods in Multiobjective Optimization," *Bull. Malays. Math. Sci. Soc.*, vol. 42, p. 1875–1905, 09 2019. doi: 10.1007/s40840-017-0579-4

[26] M. Grzegorowski, "Selected aspects of interactive feature extraction," Ph.D. dissertation, University of Warsaw, 2021.

[27] D. Granato, J. S. Santos, G. B. Escher, B. L. Ferreira, and R. M. Maggio, "Use of principal component analysis (PCA) and hierarchical cluster analysis (HCA) for multivariate association between bioactive compounds and functional properties in foods: A critical perspective," *Trends in Food Science & Technology*, vol. 72, pp. 83–90, 2018. doi: 10.1016/j.tifs.2017.12.006

[28] M. Grzegorowski and S. Stawicki, "Window-based feature extraction framework for multi-sensor data: A posture recognition case study," in *2015 Federated Conference on Computer Science and Information Systems, FedCSIS 2015, Lódz, Poland, September 13-16, 2015*, ser. Annals of Computer Science and Information Systems, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., vol. 5. IEEE, 2015. doi: 10.15439/2015F425 pp. 397–405. [Online]. Available: https://doi.org/10.15439/2015F425

[29] E. Zdravevski, P. Lameski, R. Mingov, A. Kulakov, and D. Gjorgjevikj, "Robust histogram-based feature engineering of time series data," in *2015 Federated Conference on Computer Science and Information Systems, FedCSIS 2015, Lódz, Poland, September 13-16, 2015*, ser. Annals of Computer Science and Information Systems, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., vol. 5. IEEE, 2015. doi: 10.15439/2015F420 pp. 381–388. [Online]. Available: https://doi.org/10.15439/2015F420

[30] B. Petrovska, E. Zdravevski, P. Lameski, R. Corizzo, I. Stajduhar, and J. Lerga, "Deep Learning for Feature Extraction in Remote Sensing: A Case-Study of Aerial Scene Classification," *Sensors*, vol. 20, no. 14, p. 3906, 2020. doi: 10.3390/s20143906. [Online]. Available: https://doi.org/10.3390/s20143906

[31] D. Ślęzak, A. Chadzynska-Krasowska, J. Holland, P. Synak, R. Glick, and M. Perkowski, "Scalable cyber-security analytics with a new summary-based approximate query engine," in *2017 IEEE International Conference on Big Data (IEEE BigData 2017), Boston, MA, USA, December 11-14, 2017*, J. Nie, Z. Obradovic, T. Suzumura, R. Ghosh, R. Nambiar, C. Wang, H. Zang, R. Baeza-Yates, X. Hu, J. Kepner, A. Cuzzocrea, J. Tang, and M. Toyoda, Eds. IEEE Computer Society, 2017. doi: 10.1109/BigData.2017.8258128 pp. 1840–1849. [Online]. Available: https://doi.org/10.1109/BigData.2017.8258128

[32] D. Ślęzak, R. Glick, P. Betlinski, and P. Synak, "A new approximate query engine based on intelligent capture and fast transformations of granulated data summaries," *J. Intell. Inf. Syst.*, vol. 50, no. 2, pp. 385–414, 2018. doi: 10.1007/s10844-017-0471-6. [Online]. Available: https://doi.org/10.1007/s10844-017-0471-6

[33] M. Muniswamaiah, T. Agerwala, and C. C. Tappert, "Approximate Query Processing for Big Data in Heterogeneous Databases," in *2020 IEEE International Conference on Big Data (IEEE BigData 2020), Atlanta, GA, USA, December 10-13, 2020*, X. Wu, C. Jermaine, L. Xiong, X. Hu, O. Kotevska, S. Lu, W. Xu, S. Aluru, C. Zhai, E. Al-Masri, Z. Chen, and J. Saltz, Eds. IEEE, 2020. doi: 10.1109/BigData50022.2020.9378310 pp. 5765–5767. [Online]. Available: https://doi.org/10.1109/BigData50022.2020.9378310

[34] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "LightGBM: A Highly Efficient Gradient Boosting Decision Tree," in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. Curran Associates, Inc., 2017. [Online]. Available: https://proceedings.neurips.cc/paper/2017/file/6449f44a102fde848669bdd9eb6b76fa-Paper.pdf

[35] E. Wari and W. Zhu, "A survey on metaheuristics for optimization in food manufacturing industry," *Applied Soft Computing*, vol. 46, pp. 328–343, 2016. doi: 10.1016/j.asoc.2016.04.034

[36] M. Okulewicz and J. Mandziuk, "A metaheuristic approach to solve Dynamic Vehicle Routing Problem in continuous search space," *Swarm Evol. Comput.*, vol. 48, pp. 44–61, 2019. doi: 10.1016/j.swevo.2019.03.008. [Online]. Available: https://doi.org/10.1016/j.swevo.2019.03.008

[37] M. Ulinski, A. Zychowski, M. Okulewicz, M. Zaborski, and H. Kordulewski, "Generalized Self-adapting Particle Swarm Optimization Algorithm," in *Parallel Problem Solving from Nature - PPSN XV - 15th International Conference, Coimbra, Portugal, September 8-12, 2018, Proceedings, Part I*, ser. Lecture Notes in Computer Science, A. Auger, C. M. Fonseca, N. Lourenço, P. Machado, L. Paquete, and L. D. Whitley, Eds., vol. 11101. Springer, 2018. doi: 10.1007/978-3-319-99253-2_3 pp. 29–40. [Online]. Available: https://doi.org/10.1007/978-3-319-99253-2_3

[38] M. Grzegorowski, A. Janusz, D. Ślęzak, and M. S. Szczuka, "On the Role of Feature Space Granulation in Feature Selection Processes," in *2017 IEEE International Conference on Big Data, BigData 2017, Boston, MA, USA, December 11-14, 2017*, J. Nie, Z. Obradovic, T. Suzumura, R. Ghosh, R. Nambiar, C. Wang, H. Zang, R. Baeza-Yates, X. Hu, J. Kepner, A. Cuzzocrea, J. Tang, and M. Toyoda, Eds. IEEE Computer Society, 2017. doi: 10.1109/BigData.2017.8258124 pp. 1806–1815.

[39] S. Stawicki, D. Ślęzak, A. Janusz, and S. Widz, "Decision Bireducts and Decision Reducts – A Comparison," *International Journal of Approximate Reasoning*, vol. 84, pp. 75–109, 2017.

[40] J. G. Bazan, A. Skowron, and P. Synak, "Dynamic Reducts as a Tool for Extracting Laws from Decisions Tables," in *Methodologies for Intelligent Systems, 8th International Symposium, ISMIS '94, Charlotte, North Carolina, USA, October 16-19, 1994, Proceedings*, ser. Lecture Notes in Computer Science, Z. W. Ras and M. Zemankova, Eds., vol. 869. Springer, 1994. doi: 10.1007/3-540-58495-1_35 pp. 346–355.

[41] S. H. Nguyen and M. S. Szczuka, "Feature Selection in Decision Systems with Constraints," in *Rough Sets - International Joint Conference, IJCRS 2016, Santiago de Chile, Chile, October 7-11,*

2016, Proceedings, ser. Lecture Notes in Computer Science, V. Flores, F. A. C. Gomide, A. Janusz, C. Meneses, D. Miao, G. Peters, D. Ślęzak, G. Wang, R. Weber, and Y. Yao, Eds., vol. 9920, 2016. doi: 10.1007/978-3-319-47160-0_49 pp. 537–547. [Online]. Available: https://doi.org/10.1007/978-3-319-47160-0_49

[42] K. Killamsetty, D. Sivasubramanian, G. Ramakrishnan, and R. Iyer, "Glister: Generalization based data subset selection for efficient and robust learning," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 9, pp. 8110–8118, May 2021. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/16988

[43] N. Zhai, P. Yao, and X. Zhou, "Multivariate Time Series Forecast in Industrial Process Based on XGBoost and GRU," in *2020 IEEE 9th Joint International Information Technology and Artificial Intelligence Conference (ITAIC)*, vol. 9, 2020. doi: 10.1109/ITAIC49862.2020.9338878 pp. 1397–1400.

[44] Y. Wang and X. Sherry Ni, "A XGBoost risk model via feature selection and Bayesian hyper-parameter optimization," *International Journal of Database Management Systems*, vol. 11, no. 01, p. 01–17, Feb 2019. doi: 10.5121/ijdms.2019.11101

[45] A. Janusz, A. Jamiołkowski, and M. Okulewicz, "Predicting the Costs of Forwarding Contracts: Analysis of Data Mining Competition Results," in *Proceedings of the 17th Conference on Computer Science and Intelligence Systems, FedCSIS 2022, Sofia, Bulgaria, September 4-7, 2022*. IEEE, 2022.

[46] J. G. Bazan, S. Bazan-Socha, S. Buregwa-Czuma, Ł. Dydo, W. Rząsa, and A. Skowron, "A Classifier Based on a Decision Tree with Verifying Cuts," *Fundam. Informaticae*, vol. 143, no. 1-2, pp. 1–18, 2016. doi: 10.3233/FI-2016-1300

[47] D. Ślęzak, M. Grzegorowski, A. Janusz, and S. Stawicki, "Toward interactive attribute selection with infolattices - A position paper," in *Rough Sets - International Joint Conference, IJCRS 2017, Olsztyn, Poland, July 3-7, 2017, Proceedings, Part II*, ser. Lecture Notes in Computer Science, L. Polkowski, Y. Yao, P. Artiemjew, D. Ciucci, D. Liu, D. Ślęzak, and B. Zielosko, Eds., vol. 10314. Springer, 2017. doi: 10.1007/978-3-319-60840-2_38 pp. 526–539. [Online]. Available: https://doi.org/10.1007/978-3-319-60840-2_38

[48] A. Janusz, G. Hao, D. Kaluza, T. Li, R. Wojciechowski, and D. Ślęzak, "Predicting escalations in customer support: Analysis of data mining challenge results," in *2020 IEEE International Conference on Big Data (IEEE BigData 2020), Atlanta, GA, USA, December 10-13, 2020*, X. Wu, C. Jermaine, L. Xiong, X. Hu, O. Kotevska, S. Lu, W. Xu, S. Aluru, C. Zhai, E. Al-Masri, Z. Chen, and J. Saltz, Eds. IEEE, 2020. doi: 10.1109/BigData50022.2020.9378024 pp. 5519–5526. [Online]. Available: https://doi.org/10.1109/BigData50022.2020.9378024