IBIMA Publishing
Communications of International Proceedings
https://ibimapublishing.com/p-articles/46AI/2025/4646525/
Vol. 2025 (34), Article ID 4646525

Multi-Objective Optimization of Regularized Self-Attention Regression Models Using NSGA-II: A Methodological Framework*

Maja CZYŻEWSKA

Institute of Computer and Information Systems,
Faculty of Cybernetics, Military University of Technology,
00-908 Warsaw, Kaliskiego 2 Street, Poland,
ORCID: 0009-0009-7829-8779

Correspondence should be addressed to: Maja CZYŻEWSKA, maja.czyzewska@wat.edu.pl

* Presented at the 46th IBIMA International Conference, 26-27 November 2025, Ronda, Spain

Abstract

Forecasting financial time series with deep neural networks requires models that are both expressive and well regularized. Regularized Self-Attention Regression (RSAR) is a hybrid architecture that combines LSTM, self-attention and convolutional components with explicit regularization, and has shown promising results in financial price prediction. However, its original formulation relies on a small number of manually selected hyperparameter configurations and a single-objective tuning strategy, which limits its adaptability to datasets with different volatility levels, noise structures and temporal dynamics. This paper addresses that gap by proposing a methodological framework that extends RSAR with multi-objective hyperparameter optimization based on an evolutionary genetic algorithm (NSGA-II). The framework defines a structured decision vector for key architectural and regularization parameters, formulates a bi-objective problem that jointly minimizes validation error and the train–validation gap, and outlines the optimization procedure built on non-dominated sorting, diversity preservation and elitist selection. The contribution is purely methodological: the paper provides a detailed description of the RSAR architecture, the underlying self-attention and regularization mechanisms, and their integration into a multi-objective search procedure. The resulting framework offers an implementation-ready basis for automatically adapting RSAR models to diverse financial markets, explicitly balancing predictive accuracy and overfitting risk. Experimental evaluation is intentionally omitted and will be presented in a subsequent study.

Keywords: Deep Neural Networks, LSTM, CNN, RSAR, Finance, Multi-objective optimization, NSGA-II

Introduction

Time-series forecasting in financial markets presents a unique set of challenges due to the presence of nonlinear dependencies, structural breaks, and high volatility. Traditional econometric methods often struggle when faced with such characteristics, whereas neural networks are capable of automatically extracting complex, nonlinear patterns from raw data without the need for predefined assumptions regarding model structure or data distribution. However, the effective use of deep learning models requires careful selection of architectures, hyperparameters, and mechanisms that mitigate overfitting and enhance generalization.

Recent research has increasingly focused on hybrid architectures that combine the strengths of multiple neural components. Long Short-Term Memory (LSTM) networks are effective at capturing long-term sequential dependencies, whereas Convolutional Neural Networks (CNNs) excel at identifying local patterns in sequential data. Although these hybrid models provide improvements over single-architecture approaches, they still lack the ability to dynamically distinguish between more and less relevant observations within the sequence. This limitation has motivated the introduction of attention-based mechanisms into forecasting architectures.

Cite this Article as: Maja CZYŻEWSKA, Vol. 2025 (34) “Multi-Objective Optimization of Regularized Self-Attention Regression Models Using NSGA-II: A Methodological Framework" Communications of International Proceedings, Vol. 2025 (34), Article ID 4646525, https://doi.org/10.5171/2025.4646525

The self-attention mechanism, originally introduced in the "Attention Is All You Need" architecture by Vaswani et al. (2017), allows a model to assign adaptive importance weights to different elements of an input sequence. In the context of financial time series, self-attention enables the model to emphasize critical temporal events while suppressing less informative patterns.

Zhou et al. (2020) proposed the Regularized Self-Attention Regression (RSAR) model, which integrates three components: an LSTM layer responsible for capturing long-term dependencies, a self-attention mechanism that adaptively allocates importance across time steps, and a CNN layer that extracts local features from the attention-weighted representation. Additionally, the model incorporates L1 regularization on bias terms and L2 regularization on weight matrices to reduce overfitting and increase the diversity of attention weights. This hybrid structure has demonstrated strong performance in forecasting precious metal prices.

Despite its advantages, the original RSAR implementation relies on manually selected hyperparameters and evaluates only a small number of configurations. Manual tuning restricts the model's ability to adapt to different datasets and may prevent the discovery of more effective architectural configurations. Furthermore, financial time series differ significantly across markets – currencies, indices, cryptocurrencies and commodities exhibit distinct volatility characteristics, noise levels and temporal dynamics – so a single, fixed RSAR configuration may be suboptimal when transferred to other instruments.

From a methodological perspective, two key issues remain open. Hyperparameter selection in the original RSAR study is treated as a single-objective problem driven mainly by validation error, without explicitly modelling the trade-off between predictive accuracy and overfitting. In addition, no systematic multi-objective optimization framework is provided that would allow RSAR configurations to be adapted in a principled way to datasets with different statistical properties.

To address these limitations, a multi-objective optimization approach can be employed. Hyperparameter search using evolutionary algorithms is particularly suited to high-dimensional, non-convex search spaces characteristic of deep learning architectures. The Non-dominated Sorting Genetic Algorithm II (NSGA-II) enables simultaneous optimization of multiple conflicting objectives. In the context of RSAR, such objectives may include minimizing validation error while also reducing the train-validation gap to limit overfitting. By exploring a diverse set of candidate solutions and selecting non-dominated configurations, NSGA-II provides a systematic and automated method for improving both model accuracy and generalization.

The main contributions of this paper are as follows:

1. A detailed treatment of regularization in RSAR, including L1 and L2 penalties and an additional attention regularize, and an analysis of how these components jointly affect model stability, sparsity and overfitting control.
2. A critical methodological analysis of the original RSAR hyperparameter selection procedure, showing its limitations in terms of restricted sensitivity testing, validation restricted to two closely related markets (gold and palladium), and lack of explicit control over the trade-off between accuracy and overfitting.
3. A formal reformulation of RSAR hyperparameter tuning as a bi-objective optimization problem, with a precisely defined decision vector $x = (d, attn, u, f)$, search space χ and objective vector $f(x) = (MSE_{val}(x), Gap(x))$.
4. The design of an NSGA-II–based optimization pipeline for RSAR, including the integration of Pareto-ranking and crowding-distance selection with the model training loop, providing an implementation-ready framework for future experimental studies on diverse financial datasets.

This paper presents a methodological description of the RSAR architecture and the integration of NSGA-II for automated hyperparameter optimization. The focus is placed on the theoretical foundations, architectural components, and implementation details necessary to reproduce or extend the model. Experimental results are intentionally omitted and will be presented in a subsequent study.

Notation used in this paper:

Symbol	Domain / Dimension	Explanation
N	$\mathbb{N}$	Length of the input sequence (number of time steps).

d	$\mathbb{N}$	Dimension of the LSTM hidden state (length of the feature vector x_i)
d_k	$\mathbb{N}$	Dimension of the query/key/value space in the attention mechanism (index "k" refers to key).
x_i	$\mathbb{R}^d$	LSTM hidden state (feature vector) at time step i.
$\mathbf{X}$	$\mathbb{R}^{N\times d}$	Matrix of LSTM hidden states; the i-th row is x_i^T (input sequence for the self-attention mechanism).
$\boldsymbol{Q}_i, \boldsymbol{K}_i, \boldsymbol{V}_i$	$\mathbb{R}^{d_k}$	Query, key, value vectors computed from x_i.
$\boldsymbol{W}_Q, \boldsymbol{W}_K, \boldsymbol{W}_V$	$\mathbb{R}^{d\times d_k}$	Weight matrices for queries, keys and values.
$\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}$	$\mathbb{R}^{N\times d_k}$	Matrices of queries, keys, and values (each row corresponds to one time step).
e_{ij}	$\mathbb{R}$	Raw similarity score between the query at time step i and the key at time step j.
$\boldsymbol{E}$	$\mathbb{R}^{N\times N}$	Similarity matrix: $\boldsymbol{E} = \boldsymbol{Q}\boldsymbol{K}^T$.
$\widehat{\boldsymbol{E}}$	$\mathbb{R}^{N\times N}$	Scaled similarity matrix: $\widehat{\boldsymbol{E}} = \frac{\boldsymbol{Q}\boldsymbol{K}^T}{\sqrt{d_k}}$
a_{ij}	(0,1)	Entry of the attention map after applying the activation function.
$\boldsymbol{A}$	$\mathbb{R}^{N\times N}$	Attention map: $\mathbf{A} = \sigma(\widehat{\mathbf{E}})$, where σ is the sigmoid activation.
$\boldsymbol{O}_i$	$\mathbb{R}^{d_k}$	Output vector for time step i.
$\boldsymbol{O}$	$\mathbb{R}^{N\times d_k}$	Matrix of attention-based output vectors.
$\sigma(\cdot)$	$\sigma: \mathbb{R} \to (0,1)$	Sigmoid activation used in RSAR
$\mathcal{L}(\theta)$	$\mathbb{R}$	Loss function of the model.
$\mathcal{L}_{L2}(\theta)$	$\mathbb{R}$	Loss function with L2 (Ridge) regularization.
$\mathcal{L}_{L1}(\theta)$	$\mathbb{R}$	Loss function with L1 (Lasso) regularization.
$\lVert\theta\rVert_1$	$\mathbb{R}$	L1 norm: $\lVert \boldsymbol{\theta} \rVert_{\mathbf{1}} = \sum_j \lvert \boldsymbol{\theta_j} \rvert$
$\lVert\theta\rVert_2^2$	$\mathbb{R}$	Squared L2 norm: $\lVert \boldsymbol{\theta} \rVert_{\mathbf{2}}^{\mathbf{2}} = \sum_j \boldsymbol{\theta_j^2}$
θ	-	Vector of model parameters (weights and biases) used in the loss function.
λ	$\mathbb{R}$	Regularization coefficient controlling the strength of L1/L2 penalties.
$Co(y_i, f_\theta(x_i))$	$\mathbb{R}$	Base cost function measuring prediction error for sample i.
$MSE_{train}(x)$	$\mathbb{R}$	Mean squared error on the training set for configuration x.
$MSE_{val}(x)$	$\mathbb{R}$	Mean squared error on the validation set for configuration x.
$Gap(x)$	$\mathbb{R}$	Train–validation gap: $Gap(x) = MSE_{train}(x) - MSE_{val}(x)$.
χ	$\subset \mathbb{R}^4$	Search space of admissible RSAR hyperparameter configurations.
x	$\mathbb{R}^4$	Decision vector $x = (d, attn, u, f)$ (dropout, attention regularization weight, number of LSTM units, number of CNN filters).
$f(x)$	$\mathbb{R}^2$	Vector of objective function values $(MSE_{val}(x), Gap(x))$.
Y	$\subset \mathbb{R}^2$	Set of attainable objective vectors: $Y = \{f(x) \mid x \in \chi\}$.

y, y^*	$\mathbb{R}^2$	Objective vectors used in the definition of Pareto dominance.
$P(Y)$	$\subset \mathbb{R}^2$	Pareto front – set of non-dominated points in Y.
m	$\mathbb{N}$	Number of objectives (in this work $m = 2$).
$cd(i)$	$\mathbb{R}$	Crowding distance of solution i in NSGA-II.

Background on RSAR

Self-Attention mechanism

The self-attention mechanism, originally introduced in the Transformer architecture by Vaswani et al. (2017), assigns adaptive importance weights to differ elements of an input sequence. In RSAR model, the mechanism operates on the sequence of feature vectors produced by the LSTM layer.
Let the input sequence be represented as matrix:

$$\boldsymbol{X} \in \mathbb{R}^{N\times d}, \;\; x_i \in \mathbb{R}^d$$

where the i-th row of $\boldsymbol{X}$ corresponds to the LSTM hidden state at time step i, denoted $x_i \in \mathbb{R}^d$.

For each x_i the query $\boldsymbol{Q}_i$**,** key $\boldsymbol{K}_i$ and value $\boldsymbol{V}_i$ vectors are computed using linear transformations:

$$\boldsymbol{Q} = \boldsymbol{X}\boldsymbol{W}_Q, \quad \boldsymbol{K} = \boldsymbol{X}\boldsymbol{W}_K, \quad \boldsymbol{V} = \boldsymbol{X}\boldsymbol{W}_V,$$

where $W_Q, W_K, W_V \in \mathbb{R}^{d\times d_k}$ are learnable weight matrices.
The similarity between elements of the sequence is computed using the scaled dot product:

$$\widehat{\boldsymbol{E}} = \frac{\boldsymbol{Q}\boldsymbol{K}^T}{\sqrt{d_k}}$$

with the scalar form:

$$\hat{e}_{ij} = \frac{\boldsymbol{Q}_i\boldsymbol{K}_j^T}{\sqrt{d_k}}$$

Each element $\hat{e}_{ij}$ is transformed through an activation function (sigmoid in the RSAR implementation):

$$a_{ij} = \sigma(\hat{e}_{ij}), \quad \boldsymbol{A} = \sigma(\widehat{\boldsymbol{E}})$$

The resulting attention matrix A determines how strongly each element of the sequence influences others. The output representation is then computed as:

$$\boldsymbol{O} = \boldsymbol{A}\boldsymbol{V}$$

This mechanism highlights informative time steps, allowing the model to prioritize relevant events in financial time series while downweighing less important observations.

RSAR architecture (Regularized Self-Attention Regression)

Regularized Self-Attention Regression (RSAR), proposed by Zhou et al. (2020), is a hybrid architecture designed for forecasting financial time series. The model combines three main components: an LSTM layer, a self-attention mechanism and a CNN layer. Additionally it incorporates explicit regularization of weights and attention to reduce overfitting.

The RSAR architecture enables simultaneous extraction of temporal and local features from financial data. At the first stage, an LSTM layer processes historical price sequences and captures long-term temporal dependencies. In the original study, the model was applied to gold and palladium prices, but the architecture is suitable for a broader class of financial instruments.

The sequence of hidden states generated by the LSTM layer is then passed to the self-attention mechanism. This mechanism adaptively assigns weights to different time steps of the input window, highlighting informative fragments of the sequence while suppressing less relevant observations. As a result, the model can focus on patterns and events that have the strongest impact on the predicted future values of the financial instrument.

In the original Transformer architecture, multi-head attention plays a central role. In contrast, the RSAR implementation uses single-head self-attention mechanism, which simplifies the model while preserving its ability to model global temporal relationships. The attention layer operates on the LSTM outputs and produces attention-weighted feature maps that integrate information from the entire input window with priorities defined by the learned attention weights.

In the next stage, the attention-based representation is processed by a one-dimensional convolutional layer (CNN). The convolutional filters slide over the sequence of attention-weighted vectors and detect local patterns that may not be fully captured by recurrent and attention mechanism alone. A subsequent pooling layer reduces dimensionality and emphasizes the most significant local features. This combination allows the model to exploit both long-range temporal structure and short-term local regularities present in financial time series.

The final part of the architecture consists of a fully connected (dense) layer with linear activation, which outputs a single forecast value for the given input window. The model is trained using mean squared error (MSE) loss function. To further improve generalization, dropout is applied before the output layer, randomly disabling a fraction of neurons during training and reducing risk of overfitting.

As suggested by the full name of the model – Regularized Self-Attention Regression – additional regularization is applied: L1 regularization to bias terms and L2 regularization to weight matrices. The attention mechanism is also regularized through an additional penalty term on the attention map, which prevents extreme attention distributions (e.g. a single time step dominating all others) and stabilizes training.

Regularization

Regularization is used to simplify the neural network by encouraging the model to generate low-magnitude weights. When a model is highly complex, it may tend to memorize the training data instead of learning general patterns, which leads to overfitting. As a result, the model performs well on the training set but its effectiveness decreases when analyzing new, previously unseen data.
The loss function with L2 regularization (Ridge) is defined as:

$$\mathcal{L}_{L2}(\theta) = \frac{1}{N}\sum_{i=1}^{N} Co(y_i, f_\theta(x_i)) + \lambda \parallel \theta \parallel_2^2$$

where the L2 penalty term

$$\|\theta\|_2^2 = \sum_j \theta_j^2$$

reduces the influence of excessively large weights. L2 regularization does not eliminate irrelevant features, but keeps the weights close to zero, which helps reduce overfitting.

The RSAR model also uses L1 regularization, which encourages sparse weights and often drives some parameters to exactly zero. The cost function with L1 (Lasso) regularization is expressed as:

$$\mathcal{L}_{L1}(\theta) = \frac{1}{N}\sum_{i=1}^{N} Co(y_i, f_\theta(x_i)) + \lambda \parallel \theta \parallel_1$$

where:

$$\parallel \theta \parallel_1 = \sum_j \mid \theta_j \mid$$

This mechanism simplifies the model and improves interpretability by eliminating unnecessary parameters.

In practice, the choice between L1 and L2 depends on the data and the problem characteristics, and in some cases a combination known as Elastic Net is used.

In RSAR, an additional regularizer is applied to the attention mechanism to prevent the attention distribution from collapsing onto a single time step, improving stability and reducing overfitting.

Problem statement and proposed RSAR extension

Limitations of manual hyperparameter selection in RSAR

As mentioned earlier, the Regularized Self-Attention Regression (RSAR) model proposed by Zhou et al. (2020) is a highly promising architecture for financial time-series forecasting. By combining an LSTM layer, a self-attention mechanism and a CNN component with explicit L1/L2 regularization, the model is able to capture long-term dependencies, emphasize informative time steps and extract local patterns while mitigating overfitting. In the original study, the model was applied to gold and palladium prices, but the architecture is suitable for a broader class of financial instruments.

At the same time, RSAR is a relatively complex hybrid model with multiple interacting hyperparameters. In the original work, the configuration was selected using a limited sensitivity analysis: only a few discrete settings were tested for selected hyperparameters, and the final architecture was chosen manually based on the observed performance. This approach is sufficient to demonstrate the effectiveness of the proposed model on the considered dataset, but it also introduces several limitations.

First, manual sensitivity testing explores only a small part of the high-dimensional hyperparameter space. For an architecture that combines LSTM, self-attention and CNN layers with several regularization terms, the number of possible configurations grows quickly, and many potentially better settings may remain untested. Second, the optimal configuration is likely to depend on the statistical properties of the analyzed time series. Financial instruments such as currencies, stock indices, cryptocurrencies and commodities differ in volatility, noise level and temporal structure, which means that a single, fixed RSAR configuration may be suboptimal when transferred to other markets.

Moreover, manual tuning typically focuses on a single performance indicator, such as validation error, and does not explicitly control the trade-off between prediction accuracy and overfitting. For deep models with high capacity there is a substantial risk that the architecture will fit the training data too closely, especially when hyperparameters are adjusted by trial and error.

For these reasons, in this work it is proposed to strengthen the original RSAR model not by changing its architecture, but by extending it with an automated, multi-objective hyperparameter optimization procedure. The goal is to replace manual sensitivity testing with a systematic search that can better exploit the potential of the RSAR architecture and adapt it to diverse financial time-series datasets.

Motivation for multi-objective hyperparameter optimization

Hyperparameter tuning for deep neural networks is rarely a single-objective problem. In the case of RSAR, two aspects are particularly important from a practical point of view. On the one hand, the model should achieve a low prediction error on the validation set. On the other hand, the gap between training and validation error should remain small, which indicates good generalization and reduced overfitting.

Optimizing only one of these quantities leads to an incomplete solution. Focusing solely on validation error may favor highly complex configurations that overfit the training data, while minimizing only the train-validation gap can result in models that are too simple and do not fully exploit the expressive power of the RSAR architecture. In practice, a compromise between these two criteria is needed rather than a single, manually chosen configuration.

Multi-objective optimization provides a natural framework for this type of trade-off. Instead of selecting one set of hyperparameters based on trial-and-error, the goal is to approximate a set of non-dominated solutions that offer different balances between accuracy and generalization. In this work, it is therefore proposed to treat RSAR hyperparameter tuning as a multi-objective problem and to solve it using the NSGA-II algorithm.

NSGA-II (Non-Dominated Sorting Genetic Algorithm II).

Multi-objective optimization makes it possible to optimize two or more criteria whose goals are often conflicting. In contrast to single-objective optimization, which seeks a single best solution, multi-objective optimization searches for a compromise between different objectives.

The Non-dominated Sorting Genetic Algorithm II (NSGA-II), proposed by Deb et al., is an evolutionary algorithm designed for such problems. It is based on classical genetic algorithms and uses standard operators such as selection, crossover and mutation. NSGA-II introduces three key mechanisms: non-dominated sorting, crowding distance and elitism.

Let $\chi \subset \mathbb{R}^4$denote the space of admissible decision vectors

$$x = (d, attn, u, f),$$

where d is the dropout rate, $attn$ is the attention regularization weight, u is the number of LSTM units and f is the number of CNN filters.

The vector-valued objective function is

$$f\colon \chi \rightarrow \mathbb{R}^2, f(x) = (f_1(x), f_2(x)) = (MSE_{val}(x), Gap(x)),$$

where $MSE_{val}(x)$ is the mean squared error on the validation set and $Gap(x) = MSE_{train}(x) - MSE_{val}(x)$ measures the train-validation gap.

Denote $Y = \{f(x) \mid x \in \chi\} \subset \mathbb{R}^2$ and let $y = f(x)$, $y^* = f(x^*)$.

Pareto dominance in the minimization setting is defined as

$$y \prec y^* \Leftrightarrow (\forall_{i\in\{1,2\}}\ y_i \leq y_i^*) \land (\exists_{i\in\{1,2\}}\ y_i < y_i^*).$$

The Pareto front is the set of all non-dominated points in Y:

$$P(Y) = \{y \in Y\colon \{y^* \in Y \mid y^* \prec y, y^* \neq y\} = \emptyset\}.$$

Non-dominated sorting.
This mechanism partitions the population into Pareto fronts. Solutions from the first front (rank 1) are non-dominated; after removing them, the next front (rank 2) is formed, and so on. The rank value encodes the degree of dominance – lower rank means a better position in the objective space.

Crowding distance.
Crowding distance is a measure of the density of solutions around a given individual in the objective space. For each front, solutions are sorted in ascending order for every objective. Then, for each solution (except the extreme ones), the crowding distance is computed as

$$cd(i) = \sum_{j=1}^{m} \frac{f_j^{(i+1)} - f_j^{(i-1)}}{f_j^{max} - f_j^{min}},$$

where m is the number of objectives, $f_j^{(i+1)}$ and $f_j^{(i-1)}$ are the neighboring values of the j-th objective, and f_j^{max}, f_j^{min} are the maximum and minimum values of this objective in the current front. Larger crowding distance means that a solution lies in a less populated region, which helps maintain diversity on the Pareto front.

Elitism.

Elitism ensures that the best solutions from previous generations are preserved. In NSGA-II, the parent and offspring populations are merged and then sorted by rank and crowding distance. The next generation is formed by selecting the best individuals according to these two criteria, which prevents the loss of high-quality solutions. In the RSAR+NSGA-II framework considered here, the algorithm searches the space of decision vectors

$$x = (d, attn, u, f)$$

to simultaneously minimize validation error $MSE_{val}(x)$ and the gap between training and validation error $Gap(x)$. This allows the model to balance predictive accuracy and generalisation ability by controlling both the architecture (number of LSTM units and CNN filters) and the regularisation strength (dropout rate and attention regularisation weight).

Conclusion

This paper presents a methodological framework for extending the Regularized Self-Attention Regression (RSAR) model with multi-objective hyperparameter optimization using the NSGA-II algorithm. The RSAR architecture, originally proposed by Zhou et al. (2020), is discussed in detail, with particular emphasis on the roles of the LSTM layer, self-attention mechanism and CNN component, as well as the regularization techniques that help control model complexity and reduce overfitting in financial time-series forecasting.

The limitations of manual hyperparameter selection in the original RSAR implementation are analyzed, highlighting the restricted coverage of the hyperparameter space and the dependence on a single dataset. In response to these issues, hyperparameter tuning is reformulated as a multi-objective optimization problem with two criteria: validation error and the train-validation gap. Within this formulation, NSGA-II was used to construct a systematic search procedure operating on a decision vector that includes dropout rate, attention regularization weight, the number of LSTM units and the number of CNN filters.

The proposed RSAR+NSGA-II framework replaces manual sensitivity testing with an automated, multi-objective optimization process that can better exploit the potential of the RSAR architecture and adapt it to diverse financial markets. By approximating a Pareto front of non-dominated configurations, the approach makes it possible to balance predictive accuracy and generalization rather than relying on a single, hand-crafted setup.

Experimental evaluation of the RSAR architecture has already been carried out by the author on several classes of financial instruments as part of a broader research project. However, detailed numerical results – including error metrics and model-to-model comparisons between RSAR and other deep learning baselines – are not reported here, as this paper is deliberately focused on the methodological formulation of the RSAR+NSGA-II framework and is subject to space limitations. These empirical results will be presented in a separate, application-oriented study. At a later stage, once NSGA-II experiments are conducted, further research may focus on analyzing the structure of the resulting Pareto fronts in order to better understand the trade-offs between validation error and generalization. These investigations are expected to further validate the practical usefulness of the RSAR+NSGA-II framework and to provide guidance for its deployment in real-world forecasting tasks.

Acknowledgment

The work was financed by the Military University of Technology in Warsaw, Poland as a part of the project No. UGB 531-000023-W500-22.

References

- Deb, K., Pratap, A., Agarwal, S. and Meyarivan, T. (2002) 'A fast and elitist multiobjective genetic algorithm: NSGA-II', IEEE Transactions on Evolutionary Computation, 6(2), pp. 182–197. DOI: 10.1109/4235.996017.

- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł. and Polosukhin, I. (2017) 'Attention is all you need', in Advances in Neural Information Processing Systems 30 (NeurIPS 2017), pp. 5998–6008. DOI: 10.48550/arXiv.1706.03762.

- Zhou, J., He, Z., Song, Y., Wang, H., Yang, X. and Lian, W. (2020) 'Precious metal price prediction based on deep regularization self-attention regression', IEEE Access, 8, pp. 2178–2187.