DOI 10.15622/ia.24.1.12

A. Ageev, A. Konstantinov, L. Utkin

ADA-NAF: SEMI-SUPERVISED ANOMALY DETECTION BASED ON THE NEURAL ATTENTION FOREST

Ageev A., Konstantinov A., Utkin L. **ADA-NAF: Semi-Supervised Anomaly Detection Based on the Neural Attention Forest.**

Abstract. In this study, we present a novel model called ADA-NAF (Anomaly Detection Autoencoder with the Neural Attention Forest) for semi-supervised anomaly detection that uniquely integrates the Neural Attention Forest (NAF) architecture which has been developed to combine a random forest classifier with a neural network computing attention weights to aggregate decision tree predictions. The key idea behind ADA-NAF is the incorporation of NAF into an autoencoder structure, where it implements functions of a compressor as well as a reconstructor of input vectors. Our approach introduces several technical advances. First, a proposed end-to-end training methodology over normal data minimizes the reconstruction errors while learning and optimizing neural attention weights to focus on hidden features. Second, a novel encoding mechanism leverages NAF's hierarchical structure to capture complex data patterns. Third, an adaptive anomaly scoring framework combines the reconstruction errors with the attention-based feature importance. Through extensive experimentation across diverse datasets, ADA-NAF demonstrates superior performance compared to state-of-the-art methods. The model shows particular strength in handling high-dimensional data and capturing subtle anomalies that traditional methods often do not detect. Our results validate the ADA-NAF's effectiveness and versatility as a robust solution for real-world anomaly detection challenges with promising applications in cybersecurity, industrial monitoring, and healthcare diagnostics. This work advances the field by introducing a novel architecture that combines the interpretability of attention mechanisms with the powerful feature learning capabilities of autoencoders.

Keywords: anomaly detection, random forest, attention mechanism, neural attention forest.

1. Introduction.

Anomaly detection is a critical task in data analysis, focusing on identifying rare events or observations that significantly deviate from the norm in a given system or dataset [1]. Its importance spans numerous fields, including finance, healthcare, manufacturing, and network security, where anomalies often indicate critical issues or potential threats [2]. These deviations can arise from various sources such as measurement errors, deliberate attacks, equipment malfunctions, or rare natural phenomena. Traditional anomaly detection methods, primarily based on rule-based systems or statistical techniques, often struggle with complex, high-dimensional data [3]. This limitation has led to the development of more sophisticated approaches, particularly in the realm of unsupervised and semi-supervised learning, where the algorithms learn to identify anomalies with limited or no labeled examples [4–6].

Unsupervised anomaly detection is particularly valuable when datasets lack labeled anomalies or when the types of anomalies are not well-defined. These techniques aim to learn the underlying structure and distribution of

the data, enabling the identification of instances that diverge from learned patterns [7]. The applications of such methods are wide-ranging, encompassing areas like cybersecurity, fraud detection, network monitoring, manufacturing quality control, medical diagnostics, and environmental monitoring [8, 9]. In recent years, significant advancements have been made in deep learning-based approaches to anomaly detection. Autoencoders, a type of artificial neural network, have shown remarkable success in this domain [10]. By learning to reconstruct input data, autoencoders can effectively capture underlying patterns and dependencies, allowing for the identification of anomalies based on reconstruction errors.

In this article, we propose a novel semi-supervised anomaly detection approach called ADA-NAF (Anomaly Detection Autoencoder with Neural Attention Forest). This model uniquely adapts the Neural Attention Forest [11] as an autoencoder to learn representations of normal data. We benchmark ADA-NAF against prominent techniques such as Isolation Forest (IF) [12] and its Deep extension (DIF) [13].

The primary aim of this research is to address limitations in existing methods, particularly for semi-supervised scenarios and complex data distributions. ADA-NAF leverages the strengths of neural attention mechanisms and Random Forests within an autoencoder framework, offering potential improvements in accuracy and interpretability compared to current state-of-the-art methods.

Our key contributions are:

– Introduction of ADA-NAF, a novel anomaly detection model that implements a pretraining step using Random Forest to cluster feature vectors, enabling its use on unlabeled data.

– Development of a multi-head extension to ADA-NAF, enhancing robustness through the adaptation of multi-head attention mechanisms.

– Comprehensive evaluation of ADA-NAF's performance against other models on benchmark datasets.

The paper is structured as follows: Section 2 reviews related work, Section 3 provides background on autoencoders and the Neural Attention Forest model, Sections 4 and 5 detail our proposed approach, Sections 6 and 7 present the experimental setup and results, and Section 8 concludes the paper.

2. Related works. Anomaly detection techniques. The field of anomaly detection has evolved from traditional statistical methods to more advanced machine learning and deep learning approaches. The authors in [14] provide a comprehensive survey of network anomaly detection techniques, covering statistical, classification-based, and clustering-based methods. Recent

advancements in deep learning have led to promising results in anomaly detection tasks [15, 16].

Deep learning in anomaly detection. Deep learning-based approaches have gained significant traction in anomaly detection research. These include self-supervised learning [17], One-Class Classification (OCC) [18], and specialized techniques for time series anomaly detection [19]. Chalapathy and Chawla [20] offer a thorough survey of deep learning methods for anomaly detection, highlighting their effectiveness across various domains.

Specialized applications. The versatility of deep learning in anomaly detection is evident in its application to diverse fields. For instance, the authors in [21] focus on anomaly detection in log data, while the authors [22] explore GAN-based methods. Suarez and Naval [23] investigate deep learning techniques for video anomaly detection, and Tschuchnig and Gadermayr [24] review anomaly detection methods in medical imaging, specifically for brain MRI.

Attention mechanisms in anomaly detection. Attention mechanisms, which enable models to focus on the most relevant parts of the data, have been successfully applied to anomaly detection tasks [25, 26]. Notable examples include:

– Study [28] proposes a GAN-based approach with an attention mechanism for detecting anomalies in semiconductor production sensor data.

– The authors in study [29] introduce a deep learning model with an attention mechanism for anomaly detection in vector magnetic field data.

– Paper [30] presents a graph-based anomaly detection algorithm utilizing an attention mechanism.

These attention-based models differ in their underlying architectures and data representations. The GAN-based approach [28] focuses on generating normal data patterns, [29] and [30] present different approaches to anomaly detection using attention mechanisms. The model in [29] uses a multi-layer neural network for vector magnetic field data, with its depth determined by input complexity and desired feature abstraction. In contrast, [30] introduces a graph-based algorithm, where depth refers to the number of graph convolution layers. While both utilize attention mechanisms, they differ fundamentally in data representation and processing: [29] uses vector inputs and traditional neural networks, while [30] leverages graph structures to capture relational data information, a capability that traditional deep learning models may lack. The depth and complexity of these models are tailored to their specific application domains and data types.

Autoencoder-based approaches. Autoencoders have proven particularly effective for anomaly detection in high-dimensional and

unbalanced datasets [5]. The authors in [6] explore the use of autoencoder ensembles to enhance anomaly detection accuracy. These approaches leverage the autoencoder's ability to learn compact representations of normal data, facilitating the identification of anomalies through reconstruction errors.

Random Forest in anomaly detection. The integration of attention mechanisms with Random Forests has emerged as a promising direction in anomaly detection research. Utkin and Konstantinov [32, 33] introduced the Attention-based Random Forest, which assigns attention weights to data in tree leaves using neural networks. This approach, framed within Nadaraya-Watson kernel regression [34], offers a novel perspective on combining tree-based methods with neural attention mechanisms. Our work, ADA-NAF, builds upon these foundations, particularly the Neural Attention Forest framework, to create a unique autoencoder-based model for semi-supervised anomaly detection. By integrating the strengths of Random Forests, neural attention, and autoencoder architectures, ADA-NAF aims to address the challenges of detecting anomalies in complex, high-dimensional data with limited labeled examples.

3. Preliminaries

3.1. Autoencoders for Anomaly Detection. Autoencoders are neural networks designed to learn the internal representation of data by training it on input and reconstructing itself as output [17, 35, 36]. One of the important applications of autoencoders is anomaly detection, that is, the detection of unusual or anomalous patterns in data.

Let there be training data $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N)$, where $\mathbf{x}_i \in \mathbb{R}^d$ is a d-dimensional vector, N is a count of training data. The task of anomaly detection is to detect anomalous samples for this training dataset. Let $\mathbf{y} = (y_1, y_2, \ldots, y_N)$ be the label vector, where $y_i \in \{0, 1\}$ indicates whether $\mathbf{x}_i$ is an anomaly ($y_i = 1$) or a normal pattern ($y_i = 0$). The problem of anomaly detection can be formulated as a supervised learning problem, where it is required to build a model $f : \mathbb{R}^d \to \{0, 1\}$ that will classify new samples as anomalies or normal. Autoencoders use only normal data for training.

Autoencoders are a class of neural networks that allow you to model non-linear dependencies and extract important features from input data. They consist of two main components: an encoder and a decoder. The encoder transforms the input data into an internal representation called the encoding, and the decoder restores the data from the encoding back to its original space.

Mathematically, let $\mathbf{x} \in \mathbb{R}^d$ be an input vector of dimension d and $\mathbf{z} \in \mathbb{R}^h$ be an encoding vector of dimension h, where $h < d$. The encoder is modeled as a function $E : \mathbb{R}^d \to \mathbb{R}^h$ and the decoder is modeled as a function

$D : \mathbb{R}^h \to \mathbb{R}^d$. Then the process of encoding and decoding can be written as follows:

$$\mathbf{z} = E(\mathbf{x}), \quad \hat{\mathbf{x}} = D(\mathbf{z}), \tag{1}$$

where $\hat{\mathbf{x}} \in \mathbb{R}^d$ is the reconstruction of the input vector $\mathbf{x} \in \mathbb{R}^d$.

To train the autoencoder, we use a recovery method that minimizes the recovery error between the input data and its reconstruction. Denote the loss function as $L(\mathbf{x}, \hat{\mathbf{x}})$, which measures the discrepancy between the input vector and its reconstruction. Popular loss functions are root mean square error (MSE) and binary cross entropy (BCE). The goal is to minimize this loss function.

In the context of anomaly detection, autoencoders can be used to detect abnormal patterns based on differences between normal and abnormal data. Training the autoencoder on a set of only normal samples allows the model to learn the characteristics of the normal data and create a model that will have a high reconstruction error for the anomalies since the anomalies will differ from the expected normal distribution.

One of the popular approaches to anomaly detection using autoencoders is the threshold approach. After training the autoencoder on normal data, we can use it to reconstruct new samples and calculate the reconstruction error. We then set a threshold ϵ above which samples are considered anomalous. Samples for which the reconstruction error exceeds the threshold are considered anomalies.

Formally, let $\mathbf{x}_{\text{test}} \in \mathbb{R}^d$ be a new sample, and its reconstruction is denoted as $\hat{\mathbf{x}}_{\text{test}} \in \mathbb{R}^d$. Then the anomaly detection algorithm can be written as follows:

$$\text{Anomaly Score}(\mathbf{x}_{\text{test}}) = L(\mathbf{x}_{\text{test}}, \hat{\mathbf{x}}_{\text{test}}). \tag{2}$$

If Anomaly Score$(\mathbf{x}_{\text{test}}) > \tau$, where $\tau \in \mathbb{R}$ is the given threshold, then sample $\mathbf{x}_{\text{test}}$ is considered anomalous.

However, the threshold approach has its limitations, as determining the optimal threshold can be challenging. Several methods can be employed to address this issue:

– Statistical methods: using measures of the reconstruction error.

– Percentile-based approach: setting the threshold at a specific percentile (e.g., 95th or 99th) of the error distribution.

– ROC curve analysis: optimizing the threshold using labeled data to maximize a chosen metric.

– Cross-validation: determining a robust threshold that generalizes across data subsets.

– Adaptive thresholding: implementing dynamic thresholds adjusting to recent data patterns.

– Ensemble methods: combining multiple thresholds for a more robust decision boundary.

The choice of method depends on the application context, available data, and the relative costs of false positives versus false negatives.

3.2. The Neural Attention Forest. The Neural Attention Forest (NAF) is a novel approach that integrates the attention mechanism into the Random Forest [11]. The primary objective is to assign attention weights, computed by neural networks of a specific architecture, to data in the leaves of decision trees and to the Random Forest itself. This is achieved within the framework of the Nadaraya-Watson kernel regression.

The attention mechanism in NAF is implemented by two distinct parts of the neural network:

1. **Tree-specific Attention**. The first part consists of neural networks with shared weights, trained for all trees. This part computes the attention weights for data in the leaves. For each tree, the attention operation is implemented as:

$$\mathbf{A}_k(\mathbf{x}) = \sum_{j \in J_k(\mathbf{x})} \alpha(\mathbf{x}, \mathbf{x}_j, \theta)\mathbf{x}_j, \tag{3}$$

$$B_k(\mathbf{x}) = \sum_{j \in J_k(\mathbf{x})} \alpha(\mathbf{x}, \mathbf{x}_j, \theta)y_j, \tag{4}$$

where $\mathbf{A}_k(\mathbf{x}) \in \mathbb{R}^d$, $B_k(\mathbf{x}) \in \mathbb{R}$, $\mathbf{x} \in \mathbb{R}^d$ and $J_k(\mathbf{x})$ represents the set of indices for which the feature vectors $\mathbf{x}_j \in \mathbb{R}^d$ fall into the same leaf of the k-th tree as $\mathbf{x} \in \mathbb{R}^d$, $y_j \in \mathbb{R}$ is the output corresponding to the feature vector $\mathbf{x}_j$, θ is a parameter of neural network. The attention weight $\alpha(\mathbf{x}, \mathbf{x}_j, \theta) \in \mathbb{R}$ is calculated by a neural network with θ parameters.

2. **Global Attention**. The second part of the neural network aggregates all the keys $\mathbf{A}_k(\mathbf{x})$ and values $B_k(\mathbf{x})$ from the tree-specific attention. The global attention operation is:

$$\hat{y} = \sum_{k=1}^{T} \beta(\mathbf{x}, \mathbf{A}_k(\mathbf{x}), \psi)B_k(\mathbf{x}), \tag{5}$$

where $\mathbf{A}_k(\mathbf{x})$ and $B_k(\mathbf{x})$ are the keys and values, respectively, computed for each tree, $\beta(\mathbf{x}, \mathbf{A}_k(\mathbf{x}), \psi) \in \mathbb{R}$ is the global attention weight calculated by a neural network with ψ params, and T is the total number of trees in the Random Forest.

It should be noted that the neural network here has a specific architecture with scaled dot-product score output to implement the attention mechanism. The first part computes the trainable attention weights for each tree's data. The second part then aggregates the weighted outputs to produce the final prediction.

4. ADA-NAF for Anomaly Detection. ADA-NAF model uniquely integrates the Neural Attention Forest framework into an autoencoder architecture for anomaly detection. The key idea is to leverage ADA-NAF as an autoencoder that compresses input vectors into encoded feature representations, and then reconstructs the original input.

A schematic depiction of leveraging ADA-NAF for anomaly detection is presented in Figure 1.

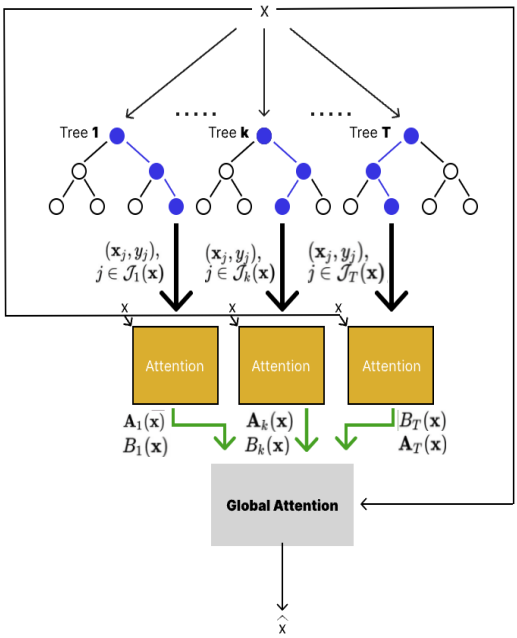

Fig. 1. The architecture of the proposed ADA-NAF model for anomaly detection

The input feature vector $\mathbf{x} \in \mathbb{R}^d$ passes through the Random Forest component to compute leaf node assignments $J_k(\mathbf{x})$ and attention-weighted aggregate representations $\mathbf{A}_k(\mathbf{x})$, $B_k(\mathbf{x})$ for each k tree. The global attention module uses these to produce a reconstructed vector $\hat{\mathbf{x}} \in \mathbb{R}^d$:

$$\hat{\mathbf{x}} = \sum_{k=1}^{T} \beta(\mathbf{x}, \mathbf{A}_k(\mathbf{x}), \psi) \cdot \mathbf{A}_k(\mathbf{x}). \tag{6}$$

The distance between the input $\mathbf{x}$ and reconstruction $\hat{\mathbf{x}}$ is computed, such as the Euclidean distance:

$$D(\mathbf{x}, \hat{\mathbf{x}}) = ||\mathbf{x} - \hat{\mathbf{x}}||_2. \tag{7}$$

If $D(\mathbf{x}, \hat{\mathbf{x}}) > \tau$, where τ is a threshold, then $\mathbf{x}$ is flagged as an anomaly. The threshold τ can be determined based on the distribution of distances for known non-anomalous data or through cross-validation.

The neural attention focuses on modeling normal data during training. At test time, anomalies result in larger reconstruction errors, allowing their detection. The key differentiating aspects from NAF are

– end-to-end training solely over normal data samples by minimizing reconstruction error;

– discovering a compressed feature encoding rather than performing supervised prediction;

– detecting anomalies based on deviation between inputs and reconstructions.

These structural modifications reshape the purpose of attention – instead of predictive performance, the focus is shifted towards the characterization of normality and subsequent identification of violations manifesting as anomalies at test time. This repurposing of the neural attention framework is the core innovation in ADA-NAF.

The general approach to training ADA-NAF for anomaly detection is shown in Figure 2:

1. Training the Random Forest component on a small labeled dataset:

– Dataset $\mathbf{X}_{labled} \in \mathbb{R}^{n \times d}$ contains n examples with dimension d with normal/anomalous class labels.

– Classification loss like cross-entropy is minimized:

$$\min_{\theta_{tree}} \mathcal{L}(\theta_{tree}) = \min_{\theta_{tree}} Loss_{CE}(\theta_{tree}, \mathbf{X}_{labled}, \mathbf{y}_{labled}), \tag{8}$$

where $\mathbf{y}_{labled} \in \mathbb{R}^n$ are labels from dataset $\mathbf{X}_{labled}$.

– The trained Random Forest is used to compute $J_k(\mathbf{x})$ - leaf indices for input vector $\mathbf{x}$.

2. Training ADA-NAF model parameters θ, ψ by minimizing reconstruction error on normal training set $\mathbf{X_{train}}$:

– X_{train} contains only normal class examples.
– Mean squared reconstruction error is minimized:

$$L(\theta, \psi) = \frac{1}{n} \sum_{\mathbf{x} \in \mathbf{X}_{train}} ||\mathbf{x} - \hat{\mathbf{x}}||^2. \tag{9}$$

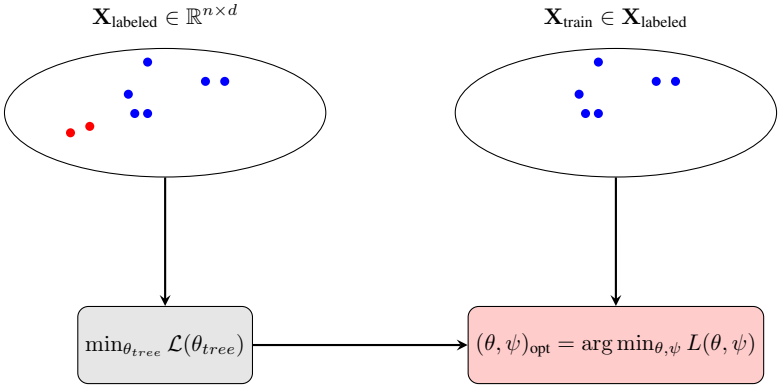

Fig. 2. ADA-NAF model training scheme for anomaly detection. Examples of normal data are shown in blue, and abnormal data in red. First, the Random Forest component is trained on a small labeled dataset. The entire ADA-NAF model is then trained to minimize the autoencoder reconstruction error using normal class examples only

The complete training and anomaly detection process for ADA-NAF is summarized in Listing 1. This algorithm outlines the two main stages of our approach: first, training the Random Forest component on a labeled subset of the data, and then training the whole ADA-NAF model on the normal data to minimize reconstruction error.

```
Input: X_train, X_test, number of trees T
Output: Anomaly scores

// Train the Random Forest
Train the Random Forest on a labeled subset of X_train
for each tree k in 1 to T:
```

```
    Compute leaf node assignments J_k(x) for all x in X_train

// Train ADA-NAF
Initialize neural attention weights θ, ψ
while not converged:
    for each x in X_train:
        Compute A_k(x) and B_k(x) using Eqs. (3) and (4)
        Compute reconstruction x̂ using Eq. (5)
        Update θ, ψ to minimize ||x - x̂||^2

// Detect anomalies
for each x in X_test:
    Compute reconstruction x̂
    Compute anomaly score as ||x - x̂||^2
```

Listing 1. Pseudocode of ADA-NAF training and anomaly detection algorithm

So the Random Forest component is first trained on labeled data, then the whole ADA-NAF model is fitted on unlabeled normal data by minimizing the autoencoder reconstruction error.

The neural network attention weights focus on normal examples during training. At test time, anomalies are then unlikely to be properly reconstructed, leading to larger errors that allow their detection.

Key advantages of the ADA-NAF anomaly detection approach include:

– Handles tabular data effectively through the Random Forest base model which stratifies the feature space.

– Learns a rich latent representation that captures boundaries between normal and anomalous data patterns.

– Leverages an initial labeled set to pre-train the Random Forest component for feature space stratification, then allows semi-supervised learning on unlabeled data for attention tuning.

– Provides local example-based explanations by selecting training points most influential for reconstructions based on learned attention weights. ADA-NAF's attention mechanisms assign importance weights to training examples. By analyzing the nearest neighbors of reconstructions according to these attention weights, we can extract influential examples that inform the model's predictions.

– Complementary to existing methods with competitive performance across various benchmark datasets.

We experimentally evaluate the proposed ADA-NAF anomaly detection method in Section 6.

5. Multi-Head ADA-NAF for Anomaly Detection. In the Multi-Head ADA-NAF (ADA-NAF-MH), we enhance the model by introducing multiple heads, as commonly done in transformer architectures [37]. Each attention head operates independently, with its own set of parameters, allowing for diverse attention patterns and reconstructions. As introduced in [37], multi-head attention projects the inputs into multiple subspaces and applies separate attention layers in parallel, before concatenating the outputs. This multi-head approach aims to capture complementary representations of the data, improving the model's robustness and accuracy.

Formally, for a given number of heads H, each head i is initialized with its own set of parameters $\theta^{(i)}$ and $\psi^{(i)}$.

Each head computes its own attention vectors as

$$\mathbf{A}_k^{(i)}(\mathbf{x}) = \sum_{j \in J_k(\mathbf{x})} \alpha^{(i)}(\mathbf{x}, \mathbf{x}_j, \theta^{(i)}) \mathbf{x}_j, \tag{10}$$

$$\mathbf{B}_k^{(i)}(\mathbf{x}) = \sum_{j \in J_k(\mathbf{x})} \alpha^{(i)}(\mathbf{x}, \mathbf{x}_j, \theta^{(i)}) y_j, \tag{11}$$

where $\alpha^{(i)}$ represents the attention weight computed by the i-th head.

The reconstruction for each head is given by:

$$\hat{\mathbf{x}}^{(i)} = \sum_{k=1}^{T} \beta^{(i)}(\mathbf{x}, \mathbf{A}_k^{(i)}(\mathbf{x}), \psi^{(i)}) \mathbf{B}_k^{(i)}(\mathbf{x}). \tag{12}$$

To aggregate the outputs of all heads, we compute an unweighted average:

$$\hat{\mathbf{x}} = \frac{1}{H} \sum_{i=1}^{H} \hat{\mathbf{x}}^{(i)}. \tag{13}$$

The model is trained to minimize the reconstruction loss on the normal training data:

$$L(\theta^{(1)}, ..., \theta^{(H)}, \psi^{(1)}, ..., \psi^{(H)}) = \sum_{\mathbf{x} \in X_{\text{train}}} ||\mathbf{x} - \hat{\mathbf{x}}||^2. \tag{14}$$

Specifically for anomaly detection, the multi-head extension enhances flexibility in capturing boundaries between normal and anomalous data points. Crucially, edge cases might be flagged by specialized heads finetuned through individual parameterizations. ADA-NAF-MHA architecture aims to fuse these signals into a unified detector of higher sensitivity.

6. ϵ-contamination attention regularization. We introduce an ϵ-contamination style regularization approach to impose robustness in the learned attention distributions while retaining sensitivity for anomaly detection. The global attention mechanism in ADA-NAF produces the reconstruction as:

$$\hat{\mathbf{x}} = \sum_{k=1}^{T} \beta(\mathbf{x}, \mathbf{A}_k(\mathbf{x}), \psi) \cdot \mathbf{A}_k(\mathbf{x}). \tag{15}$$

To enhance the robustness of this attention mechanism, we propose modifying the reconstruction process as follows:

$$\hat{\mathbf{x}}' = \sum_{k=1}^{T} ((1-\epsilon)\beta(\mathbf{x}, \mathbf{A}_k(\mathbf{x}), \psi) + \epsilon \cdot softmax(\mathbf{W})) \cdot \mathbf{A}_k(\mathbf{x}), \tag{16}$$

where $\mathbf{W} \in \mathbb{R}^T$ is a trainable parameter, randomly initialized. The contamination ratio ϵ controls the amount of mixing of $\mathbf{W}$ into the primary attention distribution β. This introduces a regularization effect that encourages the model to discover additional informative patterns beyond those identified by the original attention mechanism.

Experiments are conducted with ADA-NAF models trained using different fixed ϵ values.

We hypothesize that the ϵ-contamination attention regularization induces differing effects based on the ϵ level:

1. Small ϵ (0.1-0.2) introduces beneficial noise that improves stability and robustness to distortions without sacrificing sensitivity. Attention retains the focus on hidden features.
2. Large ϵ (>0.3) overwhelms useful signals, over-regularizing attention and reducing sensitivity to anomalies along with representations losing usefulness.
3. An optimal ϵ balances noise injection for desirable stability while avoiding dilution of attention selectivity. This optimal point is expected to be dataset and architecture-dependent.

7. Numerical experiments. The aim of this chapter is to provide a comprehensive evaluation of the proposed method using numerical experiments. The experiments are designed to demonstrate the effectiveness of the method in comparison to the other described in this article approaches, and to show the impact of various parameters on the performance of the method. In the experiments, we will compare the performance of the three models on a variety of datasets and use standard evaluation metrics such as AUC-ROC to assess the performance of each model. The results will be presented in the form of tables and graphs to allow for a clear and comprehensive comparison of the models.

Gradient descent is used for optimization with the following parameters: learning rate is 0.01, optimizer is AdamW, and the count epoch is 50.

ADA-NAF is implemented by means of software in Python. The software implementing ADA-NAF is available at https://github.com/AndreyAgeev/ada-naf.

7.1. Experimental Setup. In this section, we describe the dataset used in the experiments, the evaluation metrics, and the implementation details.

7.1.1. Datasets. We experiment over the following public anomaly detection benchmarks:

– **Arrhythmia** [38] – Collection of electrocardiogram (ECG) heartbeat segments from UCI Machine Learning Repository annotated with cardiac condition type.

– **Credit Card** [39] – Confidential credit card transactions dataset shared by a financial services company on Kaggle for analytics purposes and fraud detection.

– **Pima** [40] – Medical diagnostic measurements from Pima Indian diabetes patients released as part of open dataset collection by the National Institute for Diabetes and Digestive and Kidney Diseases.

– **Haberman** [41] – Historical dataset documenting breast cancer survival study featuring age of patients, year of surgery and number of detected axillary nodes. Hosted on UCI data repository.

– **Ionosphere** [42] – Radar data returns bounced off the ionosphere layer labeled as good or bad structures based on evidence of turbulence/stability patterns. Very common classification benchmark.

– **Seismic bumps** [43] – Recordings gathered from seismographic sensors in coal mines indicating warning signs of impending seismic bumps or just ambient tremors published as an open dataset.

– **Shuttle** [44] – System health data with sensor readings and component fault status during simulated space shuttle flights released by NASA for engineering challenges.

– **Annthyroid** [45] – Patient records tabulating biomarker readouts, test outcomes and diagnoses for differentiation of thyroid gland malfunction symptoms.

– **Bank Additional** [46] – Financial customer data on marketing campaign responses used for response modeling and fraud analysis hosted as a public dataset.

– **CelebA** [47] – Large-scale face image dataset with celebrity photos annotated for the presence/absence of multiple facial attributes like expressions, hair color, age, etc.

Table 1 shows a dataset information table.

For large real datasets, data slices are taken and instead of the full data. The slice size is indicated in the table with the description of the data. Before applying the methods, data preprocessing was carried out on some of the presented datasets, including data normalization and feature selection. The dataset preprocessing code is in https://github.com/AndreyAgeev/ada-naf.

Table 1. A brief introduction about the datasets

Dataset	normal	anomal	n feature
Arrhythmia	386	66	17
Credit	1500	400	30
Pima	500	268	8
Haberman	225	81	3
Ionosphere	225	126	33
Seismic bumps	2584	170	21
Shuttle	1000	13	9
Annthyroid	500	50	21
Bank additional	500	50	62
Celeba	500	50	39

7.1.2. Evaluation Metrics. In the experiments, we use the following evaluation metrics to assess the performance of the method:

– AUC-ROC.

To evaluate the AUC-ROC, 66.7% of the data were randomly selected for training and 33.3 % were randomly selected for testing. 33.3 % of the training dataset is also allocated to validation, which saves the best model.

7.1.3. Implementation Details. The proposed method was implemented using the programming language Python and the library PyTorch.

The following models were compared with each other:

– IF [12];

– DIF [13];

– autoencoder model with 2 hidden layers of size $d/2$ with ReLU activations;

– ADA-NAF (ADA-NAF-1) model with one hidden layer containing $d/2$ units;

– ADA-NAF (ADA-NAF-3) model with 3 hidden layers each again $d/2$ width with Tanh nonlinearities;

– multi-head ADA-NAF (ADA-MH-3-NAF-1) extending above using H=3 attention heads based on varied weight initializations (uniform, Xavier, normal distribution) that integrate both local tree-attention and global aggregation, with one hidden layer containing d/2 units;

where d is the input features for each dataset.

IF [12] is an unsupervised anomaly detection method based on the principle that anomalies are few and different, and thus should be easier to isolate in a dataset. The algorithm works by randomly selecting a feature and then randomly selecting a split value between the maximum and minimum values of that feature. This process is repeated recursively to create a tree structure. Anomalies are points that require fewer splits to be isolated from the rest of the data.

DIF [13] extends the concept of Isolation Forest by incorporating deep learning techniques. Instead of using raw features for splitting, DIF first transforms the input data using a neural network. This allows the model to learn complex, non-linear feature representations that can potentially lead to more effective isolation of anomalies. The depth of the model in DIF refers to both the depth of the neural network used for feature transformation and the depth of the isolation trees constructed on these transformed features. This combination of deep feature learning and isolation enables DIF to potentially capture more complex anomaly patterns than the original Isolation Forest.

8. Experimental Results. In this section, we present and discuss the experimental results.

8.1. Comparison between models. To measure the performance, we use the AUC-ROC score, which is a commonly used metric in anomaly detection. We compare the AUC-ROC score dependence on several datasets.

The results are shown in Table 2.

For these experiments, the number of trees was taken as 100 and the count epoch is 50. Mean squared error (MSE) was taken as a distance function for ADA-NAF-based models.

We use 3 different seeds when building trees, and 3 shuffle train/test datasets, and then average the results of the metrics. The results in Table 2 demonstrate that the proposed ADA-NAF approach is competitive with state-of-the-art anomaly detection techniques across the diverse benchmark datasets. On

the Credit Card, Ionosphere, Annthyroid, Shuttle, Celeba and Bank Additional datasets, ADA-NAF models achieve the highest or near highest AUC-ROC scores compared to the baseline methods. This underscores ADA-NAF's capability to effectively model the complex boundaries between normal and anomalous data patterns for these domains.

Table 2. Comparison of AUC-ROC for different models on different datasets

Dataset	**IF**	**DIF**	**Autoencoder**
Arrhytmia	$\mathbf{0.791 \pm 0.02}$	0.780 ± 0.01	0.771 ± 0.01
Credit	0.968 ± 0.01	0.935 ± 0.01	0.962 ± 0.01
Haberman	$\mathbf{0.658 \pm 0.08}$	0.602 ± 0.07	0.571 ± 0.07
Ionosphere	0.912 ± 0.03	0.909 ± 0.02	0.896 ± 0.03
Pima	$\mathbf{0.727 \pm 0.03}$	0.689 ± 0.03	0.689 ± 0.01
Seismic bumps	0.690 ± 0.02	$\mathbf{0.710 \pm 0.02}$	0.680 ± 0.02
Shuttle	0.842 ± 0.04	0.966 ± 0.04	0.967 ± 0.02
Annthyroid	0.848 ± 0.04	0.728 ± 0.03	0.697 ± 0.04
Celeba	0.732 ± 0.06	0.772 ± 0.05	0.740 ± 0.01
Bank additional	0.731 ± 0.03	0.717 ± 0.04	0.796 ± 0.06
Average	0.790	0.781	0.777
Dataset	**ADA-NAF-1**	**ADA-NAF-3**	**ADA-NAF-MH**
Arrhytmia	0.702 ± 0.06	0.674 ± 0.06	0.754 ± 0.04
Credit	0.972 ± 0.01	0.941 ± 0.02	$\mathbf{0.997 \pm 0.01}$
Haberman	0.580 ± 0.10	0.569 ± 0.09	0.567 ± 0.04
Ionosphere	0.960 ± 0.02	0.922 ± 0.02	$\mathbf{0.961 \pm 0.01}$
Pima	0.641 ± 0.03	0.617 ± 0.03	0.637 ± 0.03
Seismic bumps	0.695 ± 0.02	0.683 ± 0.02	0.704 ± 0.02
Shuttle	$\mathbf{0.990 \pm 0.01}$	0.790 ± 0.11	0.936 ± 0.09
Annthyroid	$\mathbf{0.891 \pm 0.02}$	0.748 ± 0.05	0.867 ± 0.03
celeba	0.697 ± 0.10	0.672 ± 0.09	$\mathbf{0.843 \pm 0.04}$
Bank additional	0.721 ± 0.03	0.700 ± 0.07	$\mathbf{0.797 \pm 0.03}$
Average	0.785	0.732	**0.806**

Notably, the DIF technique displays superior performance on the Arrhythmia, Haberman, and Pima datasets as evidenced by the higher AUC-ROC values. This indicates DIF's particular suitability for modeling anomalies in these medical-related tabular datasets. However, ADA-NAF variants still produce reasonable scores, elucidating the promise of the neural attention-based framework. An ablative analysis reveals that typically the multi-head ADA-NAF configuration demonstrates a slight edge over the single-headed version, corroborating the benefits of fusing diverse attention representations. Comparing shallow and deeper ADA-NAF models, gains from additional layers are dataset dependent – aligning with established knowledge that

optimal depth is contingent on data complexity. In summary, ADA-NAF puts forth a highly competitive semi-supervised technique for anomaly detection grounded in cutting-edge neural attention architectures. The experiments validate applicability to heterogeneous data domains with performance rivaling current state-of-the-art approaches. This underscores the potential of innovating tailored neural attention mechanisms for advancing anomaly detection.

8.2. Noise contamination of training set. Real-world datasets often contain some degree of inherent anomalies or mislabeled points overlapping with normal classes. It is vital to evaluate model robustness towards such contaminated training data. We simulate this by injecting anomalies masked as normal into the ADA-NAF training set in a controlled manner. To evaluate the impact of contaminated training data, we construct noisy variants of the normal set X_{normal} as follows:

1. We start with the completely normal training examples X_{normal} with size as $|X_{anomalous}|$, where $X_{anomalous}$ is the anomalous samples.

2. Noisy normal sets are prepared by combining normal and anomalous points:

$X_{normal}^{(A)}$: Take A% of the anomalous samples from $X_{anomalous}$ and combine them with (100 - A)% of the normal instances in X_{normal}.

3. Train ADA-NAF separately on X_{normal}, $X_{normal}^{(12.5)}$, $X_{normal}^{(25)}$, $X_{normal}^{(37.5)}$ and $X_{normal}^{(50)}$ while RF trains on balanced data.

The experiment evaluates how introducing different levels of noise into the training set affects the performance of ADA-NAF models in detecting anomalies. Understanding the impact of noise on ADA-NAF training is critical to improving model robustness and reliability, especially in real-world scenarios where the data often contains some level of noise or anomalies.

We use 3 seeds while building RF with 3 shuffle dataset cross-validation, the count epoch is 50, number of trees is 100.

Results in Figure 3 – 5 showcase the impact on AUC scores for the Ionosphere, Annthyroid and Celeba datasets. We compare 3 model variants: single hidden layer ADA-NAF, 3 hidden layer ADA-NAF and 3 headed attention with 1 hidden layer. Overall we observe performance degradation as anomalous points increasingly pollute the normal class data. The Annthyroid dataset displays the most graceful lowering of AUC compared to sudden drops for Celeba, indicating robustness. The multi-head architecture appears significantly more stable than standard ADA-NAF, retaining higher accuracy despite up to 50% contamination. This underscores the flaw of diversified attention heads in establishing reliable boundaries between normality and anomaly. The findings highlight the variability in the impact of label noise on

different models and datasets. For real-world anomaly detection, pre-filtering training data to minimize contamination would enhance ADA-NAF's detection capability. Online learning schemes to continually adapt to new normal and anomalous data are another strategy to offset declining performance over time. The analysis provides vital perspectives on the reliability of semi-supervised approaches in practice.

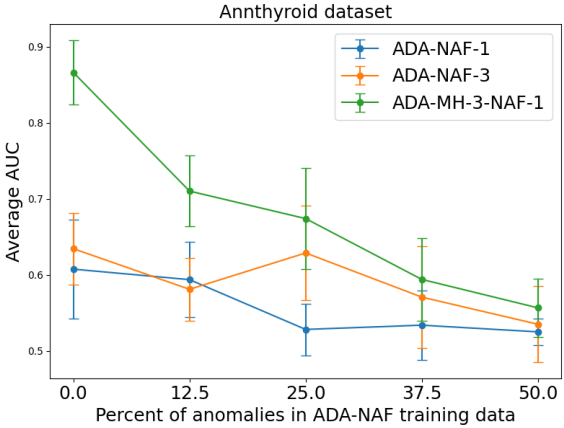

Fig. 3. AUC graphs for the Annyhyroid dataset with different noise injection

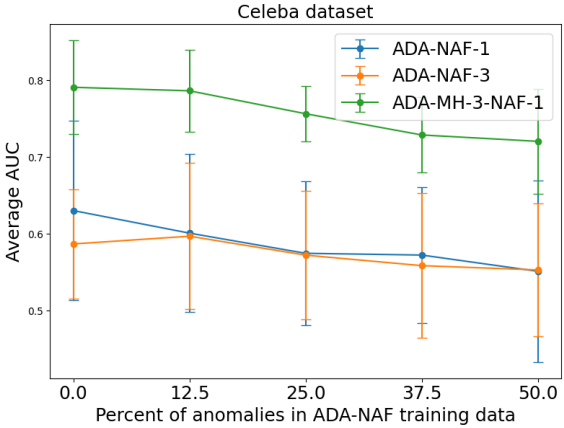

Fig. 4. AUC graphs for the Celeba dataset with different noise injection

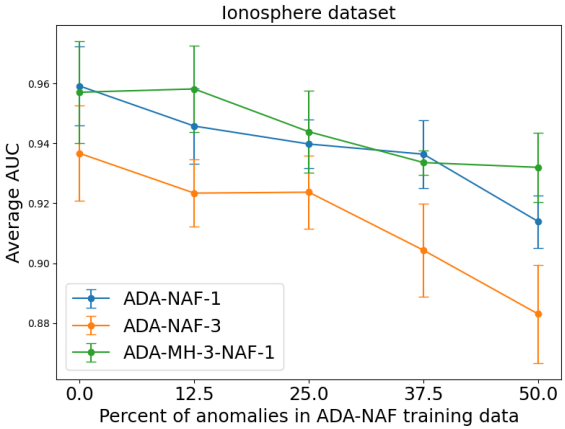

Fig. 5. AUC graphs for the Ionosphere dataset with different noise injection

8.3. ϵ-contamination. To evaluate the impact of the proposed regularization method, we conduct experiments with ADA-NAF models trained using different fixed ϵ values. We train multiple models while keeping all settings identical except for the ϵ hyperparameter controlling the admixture amount of the random attention matrix W. Models are trained end-to-end on the same normal training set for a fixed number of epochs. Specifically, we select ϵ values spread over the [0, 0.5] range. For each value, we train an ADA-NAF variant regularized with the corresponding ϵ contamination ratio. We plot the metrics vs ϵ to analyze the regularization impact. We hypothesize a non-linear influence, with small ϵ likely improving robustness hence detection accuracy by balancing noise, while large values can overwhelm true signals. The optimal ϵ is expected to be dataset-dependent.

The key findings from the experiments are:

1. In the Celeba Dataset (Figure 6), we observe that the AUC score for both ADA-NAF-1 and ADA-NAF-3 gradually decreases as the regularization parameter ϵ increases. ADA-NAF-1 demonstrates more stability, maintaining a relatively consistent AUC score. On the other hand, ADA-NAF-3 experiences a significant drop in performance, followed by a slight improvement. This pattern suggests that ADA-NAF-1, with its potentially simpler architecture, is less affected by increasing regularization, thereby indicating a steadier performance against the variations in ϵ. Conversely, the initial decline in ADA-NAF-3's performance up to a critical point $\epsilon = 0.3$ before it begins to recover slightly, underscores its vulnerability to stronger regularization effects but also hints at a possible resilience mechanism that kicks in beyond that point.

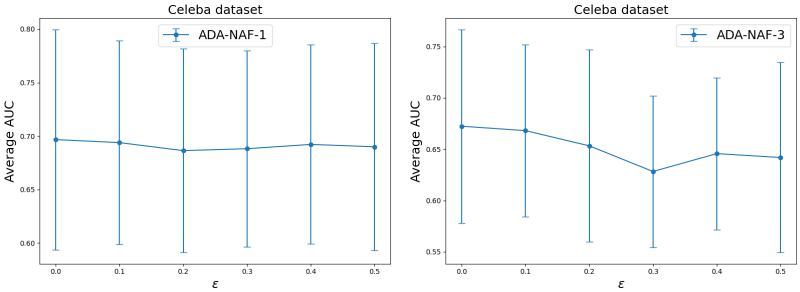

Fig. 6. AUC graphs for the Celeba dataset with different ϵ

2. Bank Additional Dataset (Figure 7). Both shallow and deeper ADA-NAF variants exhibit a peak AUC at the 0.4 contamination level. Performance is maximized with mild attention noise injection. The improvement suggests that low ϵ ratios serve more as useful perturbations rather than obstruction of meaningful attention patterns. This accords with literature on carefully tuned noise amplification improving generalization.

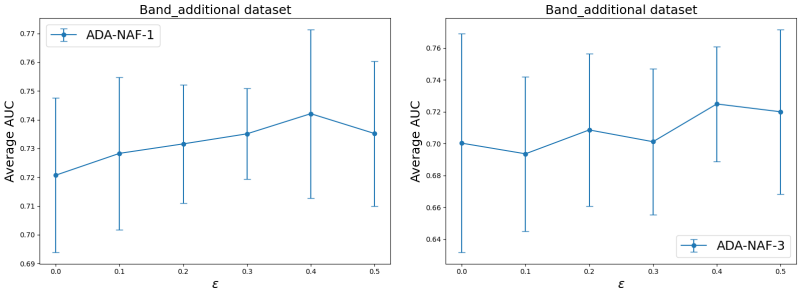

Fig. 7. AUC graphs for the Bank Additional dataset with different ϵ

3. Annthyroid Dataset (Figure 8). The model with one layer shows deterioration with increasing contamination, while the 3-layer model appears to be more stable, one can note the influence of the architecture and the choice of pollution level for different datasets.

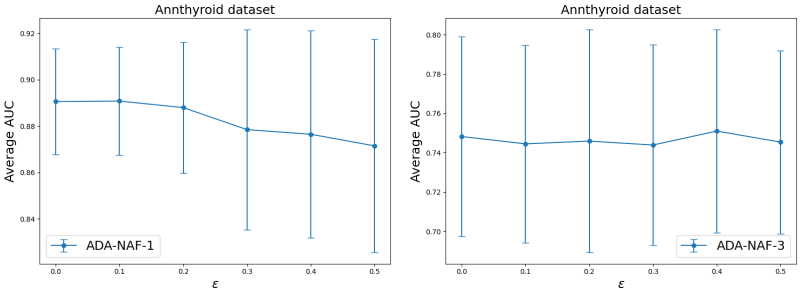

Fig. 8. AUC graphs for the Annthyroid dataset with different ϵ

9. Conclusion. This research introduced ADA-NAF - an innovative neural attention model for semi-supervised anomaly detection adapted from the Neural Attention Forest framework. We structurally modify NAF to operate as a reconstructive autoencoder powered by neural attention. Experimental results demonstrate that ADA-NAF provides a competitive approach rivaling state-of-the-art techniques over diverse anomaly detection benchmarks. The integrated architecture allows ADA-NAF to overcome the limitations of standard neural networks concerning tabular data while benefiting from the representational richness afforded by deep learning. By tuning neural attention, the model focuses intrinsically on normal data characteristics. At test time anomalies result in greater reconstruction errors enabling their detection without explicit labels. An interesting opportunity for further research includes replacing the neural network with a full-fledged transformer architecture. By tokenizing training samples into input sequences, transformers can capture complex data relationships through self-attention. This can potentially enhance the anomaly detection accuracy and interpretability of reconstructions provided by the ADA-NAF model. While our work focuses on adapting NAF for anomaly detection, the flexibility and power of this architecture suggest potential applications in various other semi-supervised learning tasks. The unique combination of Random Forests with neural attention mechanisms in NAF makes it particularly suitable for scenarios with limited labeled data. In many real-world applications across different domains, obtaining large amounts of labeled data can be expensive, time-consuming, or sometimes impossible. NAF's ability to leverage both labeled and unlabeled data effectively could prove valuable in such contexts. The model's capacity to capture complex feature interactions and its interpretability through attention weights could be beneficial in fields such as healthcare, finance, or industrial monitoring, where understanding the model's decision-making process is crucial. Furthermore,

the hierarchical structure of NAF could be advantageous in handling high-dimensional data or in tasks requiring multi-level feature extraction. While these potential applications remain to be explored, they highlight the versatility of NAF architecture and open up exciting avenues for future research beyond anomaly detection, particularly in semi-supervised learning scenarios. In conclusion, ADA-NAF contributes an interpretable semi-supervised technique to complement existing methodologies. The work highlights the importance of constructing innovative neural attention architectures tailored for anomaly detection challenges.

References

1. Chandola V., Banerjee A., Kumar V. Anomaly detection: A survey. ACM Computing Surveys. 2009. vol. 41. no. 3. pp. 1–58. DOI: 10.1145/1541880.1541882.
2. Barnett V., Lewis T. Outliers in statistical data. 3rd Edition. New York: Wiley, 1994. 608 p.
3. Grubbs F.E. Procedures for detecting outlying observations in samples. Technometrics. 1969. vol. 11. pp. 1–21. DOI: 10.1080/00401706.1969.10490657.
4. Goldstein M. Special Issue on Unsupervised Anomaly Detection. Applied Sciences. 2023. vol. 13(10). DOI: 10.3390/app13105916
5. Zhang C., Liu J., Chen W., Shi J., Yao M., Yan X., Xu N., Chen D. [Retracted] Unsupervised Anomaly Detection Based on Deep Autoencoding and Clustering. Security and Communication Networks. 2021. vol. 2021. DOI: 10.1155/2021/7389943.
6. Sarvari H., Domeniconi C., Prenkaj B., Stilo G. Unsupervised boosting-based autoencoder ensembles for outlier detection. Proceedings of Pacific-Asia Conference on Knowledge Discovery and Data Mining. 2021. pp. 91–103. DOI: 10.1007/978-3-030-75762-5_8.
7. Yoshihara K., Takahashi K. A simple method for unsupervised anomaly detection: An application to Web time series data. Plos one. 2022. vol. 17. no. 1. DOI: 10.1371/journal.pone.0262463.
8. Kiran B.R., Thomas D.M., Parakkal R. An overview of deep learning based methods for unsupervised and semi-supervised anomaly detection in videos. Journal of Imaging. 2018. vol. 4. no. 2. DOI: 10.3390/jimaging4020036.
9. Al-amri R., Murugesan R.K., Man M., Abdulateef A.F., Al-Sharafi M.A., Alkahtani A.A. A review of machine learning and deep learning techniques for anomaly detection in IoT data. Applied Sciences. 2021. vol. 11. no. 12. DOI: 10.3390/app11125320.
10. Finke T., Kramer M., Morandini A., Muck A., Oleksiyuk I. Autoencoders for unsupervised anomaly detection in high energy physics. Journal of High Energy Physics. 2021. vol. 2021. no. 6. DOI: 10.1007/JHEP06(2021)161.
11. Konstantinov A.V., Utkin L.V., Lukashin A.A., Muliukha V.A. Neural attention forests: Transformer-based forest improvement. Proceedings of International Conference on Intelligent Information Technologies for Industry. 2023. pp. 158–167.
12. Liu F.T., Kai M.T., Zhou Z.H. Isolation forest. Proceedings of 8th IEEE International Conference on Data Mining. 2008. pp. 413–422. DOI: 10.1109/ICDM.2008.17.
13. Xu H., Pang G., Wang Y., Wang Y. Deep Isolation Forest for Anomaly Detection. IEEE Transactions on Knowledge and Data Engineering. 2023. vol. 35. no. 12. pp. 12591–12604. DOI: 10.1109/TKDE.2023.3270293.
14. Ahmed M., Mahmood A.N., Hu J. A survey of network anomaly detection techniques. Journal of Network and Computer Applications. 2016. vol. 60. pp. 19–31. DOI: 10.1016/j.jnca.2015.11.016.

15. Liao Y., Bartler A., Yang B. Anomaly detection based on selection and weighting in latent space. Proceedings of 2021 IEEE 17th International Conference on Automation Science and Engineering (CASE). 2021. pp. 409–415. DOI: 10.1109/CASE49439.2021.9551267.
16. Xu J., Wu H., Wang J., Long M. Anomaly Transformer: Time Series Anomaly Detection with Association Discrepancy. Proceedings of Tenth International Conference on Learning Representations. 2022.
17. Hojjati H., Ho T.K.K., Armanfard N. Self-supervised anomaly detection: A survey and outlook. arXiv preprint arXiv:2205.05173. 2022.
18. Perera P., Oza P., Patel V.M. One-class classification: A survey. arXiv preprint arXiv:2101.03064. 2021.
19. Darban Z.Z., Webb G.I., Pan S., Aggarwal C.C., Salehi M. Deep learning for time series anomaly detection: A survey. ACM Computing Surveys. 2024. vol. 57. no. 1. DOI: 10.1145/369133.
20. Chalapathy R., Chawla S. Deep learning for anomaly detection: A survey. arXiv preprint arXiv:1901.03407. 2019.
21. Landauer M., Onder S., Skopik F., Wurzenberger M. Deep learning for anomaly detection in log data: A survey. Machine Learning with Applications. 2023. vol. 12. DOI: 10.1016/j.mlwa.2023.100470.
22. Di Mattia F., Galeone P., De Simoni M., Ghelfi E. A survey on gans for anomaly detection. arXiv preprint arXiv:1906.11632. 2019.
23. Suarez J.J.P., Naval Jr P.C. A survey on deep learning techniques for video anomaly detection. arXiv preprint arXiv:2009.14146. 2020.
24. Tschuchnig M.E., Gadermayr M. Anomaly detection in medical imaging-a mini review. Proceedings of the 4th International Data Science Conference–iDSC 2021. 2022. pp. 33–38.
25. Niu Z., Zhong G., Yu H. A review on the attention mechanism of deep learning. Neurocomputing. 2021. vol. 452. pp. 48–62. DOI: 10.1016/j.neucom.2021.03.091.
26. Bahdanau D., Cho K., Bengio Y. Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473. 2014.
27. Zhu Y., Newsam S. Motion-aware feature for improved video anomaly detection. arXiv preprint arXiv:1907.10211. 2019.
28. Hashimoto M., Ide Y., Aritsugi M. Anomaly detection for sensor data of semiconductor manufacturing equipment using a GAN. Procedia Computer Science. 2021. vol. 192. pp. 873–882. DOI: 10.1016/j.procs.2021.08.090.
29. Wu X., Huang S., Li M., Deng Y. Vector magnetic anomaly detection via an attention mechanism deep-learning model. Applied Sciences. 2021. vol. 11. no. 23. DOI: 10.3390/app112311533.
30. Yu Y., Zha Z., Jin B., Wu G., Dong C. Graph-Based Anomaly Detection via Attention Mechanism. Proceedings of on: 18th International Conference on Intelligent Computing Theories and Application. 2022. pp. 401–411. DOI: 10.1007/978-3-031-13870-6_33.
31. Tang T.W., Hsu H., Huang W.R., Li K.M. Industrial Anomaly Detection with Skip Autoencoder and Deep Feature Extractor. Sensors. 2022. vol. 22. no. 23. DOI: 10.3390/s22239327.
32. Utkin L.V., Konstantinov A.V. Attention-based random forest and contamination model. Neural Networks: the official journal of the International Neural Network Society. 2022. vol. 154. pp. 346–359.
33. Utkin L., Ageev A., Konstantinov A., Muliukha V. Improved Anomaly Detection by Using the Attention-Based Isolation Forest. Algorithms. 2023. vol. 16. no. 1. DOI: 10.3390/a16010019.

34. Cai Z. Weighted nadaraya–watson regression estimation. Statistics and probability letters. 2001. vol. 51. no. 3. pp. 307–318. DOI: 10.1016/S0167-7152(00)00172-3.
35. Rumelhart D.E., Hinton G.E., Williams R.J. Learning internal representations by error propagation. Parallel Distributed Processing: Explorations in the Microstructure of Cognition. 1986. vol. 1. pp. 318–362.
36. Hawkins S., He H., Williams G., Baxter R. Outlier detection using replicator neural networks. International Conference on Data Warehousing and Knowledge Discovery. 2002. pp. 170–180. DOI: 10.1007/3-540-46145-0_17.
37. Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A.N., Kaiser L., Polosukhin I. Attention is all you need. Advances in neural information processing systems. 2017. vol. 30.
38. Arrhythmia Dataset. Available at: https://www.kaggle.com/code/mtavares51/binary-classification-on-arrhythmia-dataset. (accessed 30.05.2024).
39. Credit Card Fraud Detection Dataset. Available at: https://www.kaggle.com/code/shivamsekra/credit-card-fraud-detection-eda-isolation-forest. (accessed 30.05.2024).
40. Pima Indians Diabetes Dataset. Available at: https://www.kaggle.com/code/hafizramadan/data-science-project-iii. (accessed 30.05.2024).
41. Haberman's Survival Dataset. Available at: https://www.kaggle.com/datasets/gilsousa/habermans-survival-data-set. (accessed 30.05.2024).
42. Ionosphere Dataset. Available at: https://www.kaggle.com/code/zymzym/classification-of-the-ionosphere-dataset-by-knn. (accessed 30.05.2024).
43. Seismic Bumps Dataset. Available at: https://www.kaggle.com/datasets/pranabroy94/seismic-bumps-data-set. (accessed 30.05.2024).
44. Shuttle Dataset. Available at: https://github.com/xuhongzuo/deep-iforest/tree/main. (accessed 30.05.2024).
45. Annthyroid Dataset. Available at: https://github.com/GuansongPang/deviation-network. (accessed 30.05.2024).
46. Bank Additional Dataset. Available at: https://github.com/GuansongPang/deviation-network. (accessed 30.05.2024).
47. CelebA Dataset. Available at: https://github.com/GuansongPang/deviation-network. (accessed 30.05.2024).

Ageev Andrey — Ph.D. student, Institute of computer science and technology, Peter the Great St. Petersburg Polytechnic University. Research interests: machine learning, bioinformatics, large language models, computer vision. The number of publications — 5. andreyageev1@mail.ru; 29, Polytechnicheskaya St., 195251, St. Petersburg, Russia; office phone: +7(812)775-0510.

Konstantinov Andrei — Ph.D. student, Institute of computer science and technology, Peter the Great St. Petersburg Polytechnic University; Assistant of the laboratory, Research laboratory of neural network technologies and artificial intelligence, Peter the Great St. Petersburg Polytechnic University. Research interests: machine learning, computer vision, image processing. The number of publications — 37. andrue.konst@gmail.com; 29, Polytechnicheskaya St., 195251, St. Petersburg, Russia; office phone: +7(911)954-5565.

Utkin Lev — Ph.D., Dr.Sci., Professor, Head of the institute, Institute of computer science and technology, Peter the Great St. Petersburg Polytechnic University; Head of the laboratory, Research laboratory of neural network technologies and artificial intelligence, Peter the Great St. Petersburg Polytechnic University. Research interests: machine learning, imprecise probability theory, decision making. The number of publications — 300. lev.utkin@gmail.com; 29, Polytechnicheskaya St., 195251, St. Petersburg, Russia; office phone: +7(921)344-6390.

Acknowledgements. The research is partially funded by the Ministry of Science and Higher Education of the Russian Federation as part of state assignments "Development and research of machine learning models for solving fundamental problems of artificial intelligence for the fuel and energy complex"(topic FSEG-2024-0027).

УДК 004.85 **DOI** 10.15622/ia.24.1.12

А.Ю. Агеев, А.В. Константинов, Л.В. Уткин

ADA-NAF: ПОЛУКОНТРОЛИРУЕМОЕ ОБНАРУЖЕНИЕ АНОМАЛИЙ НА ОСНОВЕ НЕЙРОННОГО ЛЕСА ВНИМАНИЯ

Агеев А.Ю., Константинов А.В., Уткин Л.В. **ADA-NAF: Полуконтролируемое обнаружение аномалий на основе нейронного леса внимания.**

Аннотация. В этом исследовании мы представляем новую модель под названием ADA-NAF (автоэнкодер обнаружения аномалий с нейронным лесом внимания) для полуконтролируемого обнаружения аномалий, которая уникальным образом интегрирует архитектуру нейронного леса внимания (NAF), которая была разработана для объединения случайного классификатора леса с нейронной сетью, вычисляющей веса внимания для агрегации прогнозов дерева решений. Ключевая идея ADA-NAF заключается в включении NAF в структуру автоэнкодера, где он реализует функции компрессора, а также реконструктора входных векторов. Наш подход представляет несколько технических достижений. Во-первых, предлагаемая сквозная методология обучения по обычным данным, которая минимизирует ошибки реконструкции при обучении и оптимизации нейронных весов внимания для фокусировки на скрытых признаках. Во-вторых, новый механизм кодирования, который использует иерархическую структуру NAF для захвата сложных шаблонов данных. В-третьих, адаптивная структура оценки аномалий, которая объединяет ошибки реконструкции с важностью признаков на основе внимания. Благодаря обширным экспериментам с различными наборами данных ADA-NAF демонстрирует превосходную производительность по сравнению с современными методами. Модель демонстрирует особую силу в обработке многомерных данных и выявлении тонких аномалий, которые традиционные методы часто не обнаруживают. Наши результаты подтверждают эффективность и универсальность ADA-NAF как надежного решения для реальных задач обнаружения аномалий с перспективными приложениями в кибербезопасности, промышленном мониторинге и диагностике здравоохранения. Эта работа продвигает эту область, представляя новую архитектуру, которая сочетает в себе интерпретируемость механизмов внимания с мощными возможностями обучения признакам автоэнкодеров.

Ключевые слова: обнаружение аномалий, случайный лес, механизм внимания, нейронный лес внимания.

Литература

1. Chandola V., Banerjee A., Kumar V. Anomaly detection: A survey. ACM Computing Surveys. 2009. vol. 41. no. 3. pp. 1–58. DOI: 10.1145/1541880.1541882.
2. Barnett V., Lewis T. Outliers in statistical data. 3rd Edition. New York: Wiley, 1994. 608 p.
3. Grubbs F.E. Procedures for detecting outlying observations in samples. Technometrics. 1969. vol. 11. pp. 1–21. DOI: 10.1080/00401706.1969.10490657.
4. Goldstein M. Special Issue on Unsupervised Anomaly Detection. Applied Sciences. 2023. vol. 13(10). DOI: 10.3390/app13105916
5. Zhang C., Liu J., Chen W., Shi J., Yao M., Yan X., Xu N., Chen D. [Retracted] Unsupervised Anomaly Detection Based on Deep Autoencoding and Clustering. Security and Communication Networks. 2021. vol. 2021. DOI: 10.1155/2021/7389943.
6. Sarvari H., Domeniconi C., Prenkaj B., Stilo G. Unsupervised boosting-based autoencoder ensembles for outlier detection. Proceedings of Pacific-Asia Conference on

Knowledge Discovery and Data Mining. 2021. pp. 91–103. DOI: 10.1007/978-3-030-75762-5_8.

7. Yoshihara K., Takahashi K. A simple method for unsupervised anomaly detection: An application to Web time series data. Plos one. 2022. vol. 17. no. 1. DOI: 10.1371/journal.pone.0262463.
8. Kiran B.R., Thomas D.M., Parakkal R. An overview of deep learning based methods for unsupervised and semi-supervised anomaly detection in videos. Journal of Imaging. 2018. vol. 4. no. 2. DOI: 10.3390/jimaging4020036.
9. Al-amri R., Murugesan R.K., Man M., Abdulateef A.F., Al-Sharafi M.A., Alkahtani A.A. A review of machine learning and deep learning techniques for anomaly detection in IoT data. Applied Sciences. 2021. vol. 11. no. 12. DOI: 10.3390/app11125320.
10. Finke T., Kramer M., Morandini A., Muck A., Oleksiyuk I. Autoencoders for unsupervised anomaly detection in high energy physics. Journal of High Energy Physics. 2021. vol. 2021. no. 6. DOI: 10.1007/JHEP06(2021)161.
11. Konstantinov A.V., Utkin L.V., Lukashin A.A., Muliukha V.A. Neural attention forests: Transformer-based forest improvement. Proceedings of International Conference on Intelligent Information Technologies for Industry. 2023. pp. 158–167.
12. Liu F.T., Kai M.T., Zhou Z.H. Isolation forest. Proceedings of 8th IEEE International Conference on Data Mining. 2008. pp. 413–422. DOI: 10.1109/ICDM.2008.17.
13. Xu H., Pang G., Wang Y., Wang Y. Deep Isolation Forest for Anomaly Detection. IEEE Transactions on Knowledge and Data Engineering. 2023. vol. 35. no. 12. pp. 12591–12604. DOI: 10.1109/TKDE.2023.3270293.
14. Ahmed M., Mahmood A.N., Hu J. A survey of network anomaly detection techniques. Journal of Network and Computer Applications. 2016. vol. 60. pp. 19–31. DOI: 10.1016/j.jnca.2015.11.016.
15. Liao Y., Bartler A., Yang B. Anomaly detection based on selection and weighting in latent space. Proceedings of 2021 IEEE 17th International Conference on Automation Science and Engineering (CASE). 2021. pp. 409–415. DOI: 10.1109/CASE49439.2021.9551267.
16. Xu J., Wu H., Wang J., Long M. Anomaly Transformer: Time Series Anomaly Detection with Association Discrepancy. Proceedings of Tenth International Conference on Learning Representations. 2022.
17. Hojjati H., Ho T.K.K., Armanfard N. Self-supervised anomaly detection: A survey and outlook. arXiv preprint arXiv:2205.05173. 2022.
18. Perera P., Oza P., Patel V.M. One-class classification: A survey. arXiv preprint arXiv:2101.03064. 2021.
19. Darban Z.Z., Webb G.I., Pan S., Aggarwal C.C., Salehi M. Deep learning for time series anomaly detection: A survey. ACM Computing Surveys. 2024. vol. 57. no. 1. DOI: 10.1145/369133.
20. Chalapathy R., Chawla S. Deep learning for anomaly detection: A survey. arXiv preprint arXiv:1901.03407. 2019.
21. Landauer M., Onder S., Skopik F., Wurzenberger M. Deep learning for anomaly detection in log data: A survey. Machine Learning with Applications. 2023. vol. 12. DOI: 10.1016/j.mlwa.2023.100470.
22. Di Mattia F., Galeone P., De Simoni M., Ghelfi E. A survey on gans for anomaly detection. arXiv preprint arXiv:1906.11632. 2019.
23. Suarez J.J.P., Naval Jr P.C. A survey on deep learning techniques for video anomaly detection. arXiv preprint arXiv:2009.14146. 2020.
24. Tschuchnig M.E., Gadermayr M. Anomaly detection in medical imaging-a mini review. Proceedings of the 4th International Data Science Conference–iDSC 2021. 2022. pp. 33–38.

25. Niu Z., Zhong G., Yu H. A review on the attention mechanism of deep learning. Neurocomputing. 2021. vol. 452. pp. 48–62. DOI: 10.1016/j.neucom.2021.03.091.
26. Bahdanau D., Cho K., Bengio Y. Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473. 2014.
27. Zhu Y., Newsam S. Motion-aware feature for improved video anomaly detection. arXiv preprint arXiv:1907.10211. 2019.
28. Hashimoto M., Ide Y., Aritsugi M. Anomaly detection for sensor data of semiconductor manufacturing equipment using a GAN. Procedia Computer Science. 2021. vol. 192. pp. 873–882. DOI: 10.1016/j.procs.2021.08.090.
29. Wu X., Huang S., Li M., Deng Y. Vector magnetic anomaly detection via an attention mechanism deep-learning model. Applied Sciences. 2021. vol. 11. no. 23. DOI: 10.3390/app112311533.
30. Yu Y., Zha Z., Jin B., Wu G., Dong C. Graph-Based Anomaly Detection via Attention Mechanism. Proceedings of on: 18th International Conference on Intelligent Computing Theories and Application. 2022. pp. 401–411. DOI: 10.1007/978-3-031-13870-6_33.
31. Tang T.W., Hsu H., Huang W.R., Li K.M. Industrial Anomaly Detection with Skip Autoencoder and Deep Feature Extractor. Sensors. 2022. vol. 22. no. 23. DOI: 10.3390/s22239327.
32. Utkin L.V., Konstantinov A.V. Attention-based random forest and contamination model. Neural Networks: the official journal of the International Neural Network Society. 2022. vol. 154. pp. 346–359.
33. Utkin L., Ageev A., Konstantinov A., Muliukha V. Improved Anomaly Detection by Using the Attention-Based Isolation Forest. Algorithms. 2023. vol. 16. no. 1. DOI: 10.3390/a16010019.
34. Cai Z. Weighted nadaraya–watson regression estimation. Statistics and probability letters. 2001. vol. 51. no. 3. pp. 307–318. DOI: 10.1016/S0167-7152(00)00172-3.
35. Rumelhart D.E., Hinton G.E., Williams R.J. Learning internal representations by error propagation. Parallel Distributed Processing: Explorations in the Microstructure of Cognition. 1986. vol. 1. pp. 318–362.
36. Hawkins S., He H., Williams G., Baxter R. Outlier detection using replicator neural networks. International Conference on Data Warehousing and Knowledge Discovery. 2002. pp. 170–180. DOI: 10.1007/3-540-46145-0_17.
37. Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A.N., Kaiser L., Polosukhin I. Attention is all you need. Advances in neural information processing systems. 2017. vol. 30.
38. Arrhythmia Dataset. Available at: https://www.kaggle.com/code/mtavares51/binary-classification-on-arrhythmia-dataset. (accessed 30.05.2024).
39. Credit Card Fraud Detection Dataset. Available at: https://www.kaggle.com/code/shivamsekra/credit-card-fraud-detection-eda-isolation-forest. (accessed 30.05.2024).
40. Pima Indians Diabetes Dataset. Available at: https://www.kaggle.com/code/hafizramadan/data-science-project-iii. (accessed 30.05.2024).
41. Haberman's Survival Dataset. Available at: https://www.kaggle.com/datasets/gilsousa/habermans-survival-data-set. (accessed 30.05.2024).
42. Ionosphere Dataset. Available at: https://www.kaggle.com/code/zymzym/classification-of-the-ionosphere-dataset-by-knn. (accessed 30.05.2024).
43. Seismic Bumps Dataset. Available at: https://www.kaggle.com/datasets/pranabroy94/seismic-bumps-data-set. (accessed 30.05.2024).

44. Shuttle Dataset. Available at: https://github.com/xuhongzuo/deep-iforest/tree/main. (accessed 30.05.2024).
45. Annthyroid Dataset. Available at: https://github.com/GuansongPang/deviation-network. (accessed 30.05.2024).
46. Bank Additional Dataset. Available at: https://github.com/GuansongPang/deviation-network. (accessed 30.05.2024).
47. CelebA Dataset. Available at: https://github.com/GuansongPang/deviation-network. (accessed 30.05.2024).

Агеев Андрей Юрьевич — аспирант, институт компьютерных наук и технологий, Санкт-Петербургский политехнический университет Петра Великого (СПбПУ). Область научных интересов: машинное обучение, биоинформатика, большие языковые модели и компьютерное зрение. Число научных публикаций — 5. andreyageev1@mail.ru; улица Политехническая, 29, 195251, Санкт-Петербург, Россия; р.т.: +7(812)775-0510.

Константинов Андрей Владимирович — аспирант, институт компьютерных наук и технологий, Санкт-Петербургский политехнический университет Петра Великого (СПбПУ); ассистент лаборатории, научно-исследовательская лаборатория нейросетевых технологий и искусственного интеллекта, Санкт-Петербургский политехнический университет Петра Великого (СПбПУ). Область научных интересов: машинное обучение, компьютерное зрение, обработка изображений. Число научных публикаций — 37. andrue.konst@gmail.com; улица Политехническая, 29, 195251, Санкт-Петербург, Россия; р.т.: +7(911)954-5565.

Уткин Лев Владимирович — д-р техн. наук, профессор, директор института, институт компьютерных наук и технологий, Санкт-Петербургский политехнический университет Петра Великого (СПбПУ); руководитель лаборатории, научно-исследовательская лаборатория нейросетевых технологий и искусственного интеллекта, Санкт-Петербургский политехнический университет Петра Великого (СПбПУ). Область научных интересов: машинное обучение, теория неточных вероятностей, принятие решений. Число научных публикаций — 300. lev.utkin@gmail.com; улица Политехническая, 29, 195251, Санкт-Петербург, Россия; р.т.: +7(921)344-6390.

Поддержка исследований. Исследования частично финансируются Министерством науки и высшего образования РФ в рамках государственного задания «Разработка и исследование моделей машинного обучения для решения фундаментальных задач искусственного интеллекта для ТЭК» (тема ФСЭГ-2024-0027).