informatics

MDPI

Article

LARF: Two-Level Attention-Based Random Forests with a Mixture of Contamination Models

Andrei Konstantinov †, **Lev Utkin** † **and Vladimir Muliukha** *,†

Higher School of Artificial Intelligence, Peter the Great St.Petersburg Polytechnic University, Polytechnicheskaya, 29, 195251 St. Petersburg, Russia

* Correspondence: vladimir.muliukha@spbstu.ru

† These authors contributed equally to this work.

Abstract: This paper provides new models of the attention-based random forests called LARF (leaf attention-based random forest). The first idea behind the models is to introduce a two-level attention, where one of the levels is the "leaf" attention, and the attention mechanism is applied to every leaf of trees. The second level is the tree attention depending on the "leaf" attention. The second idea is to replace the softmax operation in the attention with the weighted sum of the softmax operations with different parameters. It is implemented by applying a mixture of Huber's contamination models and can be regarded as an analog of the multi-head attention, with "heads" defined by selecting a value of the softmax parameter. Attention parameters are simply trained by solving the quadratic optimization problem. To simplify the tuning process of the models, it is proposed to convert the tuning contamination parameters into trainable parameters and to compute them by solving the quadratic optimization problem. Many numerical experiments with real datasets are performed for studying LARFs. The code of the proposed algorithms is available.

Keywords: attention mechanism; random forest; Nadaraya–Watson regression; quadratic programming; contamination model

Citation: Konstantinov, A.; Utkin, L.; Muliukha, V. LARF: Two-Level Attention-Based Random Forests with a Mixture of Contamination Models. *Informatics* **2023**, *10*, 40. https://doi.org/10.3390/informatics10020040

Academic Editor: Antony Bryant

Received: 11 December 2022
Revised: 14 April 2023
Accepted: 21 April 2023
Published: 28 April 2023

1. Introduction

Several crucial improvements in neural networks have been made in recent years. One of them is the attention mechanism, which has significantly improved classification and regression models in many machine learning areas, including the natural language processing models, computer vision, etc. [1–5]. The idea behind the attention mechanism is to assign weights to features or examples in accordance with their importance and their impact on the model predictions. The attention weights are learned by incorporating an additional feedforward neural network within a neural network architecture. Additionally, the success of the attention models as components of the neural network motivates one to extend this approach to other machine learning models different from neural networks, for example, to random forests (RFs) [6]. Following this idea, a new model called the attention-based random forest (ABRF) has been developed [7,8]. This model incorporates the attention mechanism into ensemble-based models such as RFs and the gradient boosting machine [9,10]. The ABRF models stem from the interesting interpretation [1,11] of the attention mechanism through the Nadaraya–Watson kernel regression model [12,13]. The Nadaraya–Watson regression model learns a non-linear function by using a weighted average of data using a specific normalized kernel as a weighting function. A detailed description of the model can be found in Section 3. According to [7,8], attention weights in the Nadaraya–Watson regression are assigned to decision trees in an RF depending on examples which fall into leaves of trees. Weights in ABRF have trainable parameters and use Huber's ϵ-contamination model [14] for defining the attention weights. Huber's ϵ-contamination model can be regarded as a set of convex combinations of probability distributions, where one of the distributions is considered as a set of trainable attention

parameters. A detailed description of the model is provided in Section 5.1. In accordance with the ϵ-contamination model, each attention weight consists of two parts: the softmax operation with the tuning coefficient $1 - \epsilon$ and the trainable bias of the softmax weight with coefficient ϵ. One of the improvements of ABRF, which has been proposed in [15], is based on joint incorporating self-attention and attention mechanisms into the RF. The proposed models outperform ABRF, but this outperformance is not sufficient, because this model provided inferior results for several datasets. Therefore, we propose a set of models which can be regarded as extensions of ABRF and which are based on two main ideas.

The first idea is to introduce a two-level attention, where one of the levels is the "leaf" attention, i.e., the attention mechanism is applied to every leaf of a tree. As a result, we obtain the attention weights assigned to leaves and the attention weights assigned to trees. The attention weights of trees depend on the corresponding weights of leaves which belong to these trees. In other words, the attention at the second level depends on the attention at the first level, i.e., we obtain the attention of the attention. Due to the "leaf" attention, the proposed model will be abbreviated as LARF (leaf attention-based random forest).

One of the peculiarities of LARFs is using a mixture of Huber's ϵ-contamination models instead of the single contamination model, as has been conducted in ABRF. This peculiarity stems from the second idea behind the model, which takes into account the softmax operation with different parameters. In fact, we replace the standard softmax operation by the weighted sum of the softmax operations with different parameters. With this idea, we achieve two goals. First of all, we partially solve the problem of the tuning parameters of the softmax operations which are a part of the attention operations. Each value of the tuning parameter from the predefined set (from the predefined grid) is used in a separate softmax operation. Then, weights of the softmax operations in the sum are trained jointly while training other parameters. This approach can also be interpreted as the linear approximation of the softmax operations with trainable weights and with different values of tuning parameters. However, a more interesting goal is that some analogs of the multi-head attention [16] are implemented by using the mixture of contamination models, where "heads" are defined by selecting a value of the corresponding softmax operation parameter.

Additionally, in contrast to ABRF [8], where the contamination parameter ϵ of Huber's model was a tuning parameter, the LARF model considers this parameter as the training one. This allows us to significantly reduce the model tuning time and avoid the enumeration of the parameter values in accordance with the grid. The same is implemented for the mixture of the Huber's models.

Different configurations of LARF produce a set of models, which depend on trainable and tuning parameters of the two-level attention and on algorithms for their calculation.

We investigate two types of RFs in the experiments: original RFs and Extremely Randomized Trees (ERT) [17]. According to [17], the ERT algorithm chooses a split point randomly for each feature at each node and then selects the best split among these. In contrast to ERTs, original RFs choose the most optimal (not random) split of a set of features at each node in accordance with a criterion, for example, with the Gini impurity [6].

Our contributions can be summarized as follows:

1. We propose new two-level attention-based RF models, where the attention mechanism at the first level is applied to every leaf of trees, the attention at the second level incorporates the "leaf" attention, and it is applied to trees. The training of the two-level attention is reduced to solving the standard quadratic optimization problem.
2. A mixture of Huber's ϵ-contamination models is used to implement the attention mechanism at the second level. The mixture allows us to replace a set of tuning attention parameters (the temperature parameters of the softmax operations) with trainable parameters, whose optimal values are computed by solving the quadratic optimization problem. Moreover, this approach can be regarded as an analog of the multi-head attention.

3. We propose an approach to convert the tuning contamination parameters (ϵ parameters) in the mixture of the ϵ-contamination models into trainable parameters. Their optimal values are also computed by solving the quadratic optimization problem.

Many numerical experiments with real datasets are performed for studying LARFs. They demonstrate outperforming results of some LARF modifications. The code of the proposed algorithms can be found at https://github.com/andruekonst/leaf-attention-forest (accessed on 20 April 2023).

This paper is organized as follows. The related work can be found in Section 2. A brief introduction to the attention mechanism as the Nadaraya–Watson kernel regression is given in Section 3. A general approach to incorporating the two-level attention mechanism into the RF is provided in Section 4. Ways for the implementation of the two-level attention mechanism and constructing several attention-based models by using the mixture of Huber's e-contamination models are considered in Section 5. Numerical experiments with real data, which illustrate properties of the proposed models, are provided in Section 6. The concluding remarks can be found in Section 7.

2. Related Work

Attention mechanism. Due to the great efficiency of machine learning models with the attention mechanisms, different attention-based models have become of great interest in recent years. Consequently, numerous attention models have been proposed to improve the performance of machine learning algorithms. The most comprehensive analysis and description of various attention-based models can be found in in-depth surveys [1–5,18].

It is important to note that parametric attention models as parts of neural networks are mainly trained by applying the gradient-based algorithms which lead to computational problems, when training is carried out through the softmax function. Many approaches have been proposed to cope with this problem. A large part of the approaches is based on the linear approximation of the softmax attention [19–22]. Another part of the approaches is based on random feature methods to approximate the softmax function [18,23].

Another improvement of the attention-based models is to use the self-attention which was proposed in [16] as a crucial component of neural networks called Transformers. The self-attention models have also been studied in surveys [4,24–28]. This is only a small part of all the works devoted to attention and self-attention mechanisms.

It should be noted that the aforementioned models are implemented as neural networks, and they have not been studied for applications to other machine learning models, for example, to RFs. Attempts to incorporate the attention and self-attention mechanisms into the RF and the gradient boosting machine were made in [7,8,15]. Following these research works, we extend the proposed models to improve the attention-based models. Apart from this, we propose the attention models, which do not use the gradient-based algorithms for computing optimal attention parameters. The training process of the models is based on solving standard quadratic optimization problems.

Weighted RFs. A lot of approaches have been proposed in recent years to improve RFs. One of the important approaches is based on the assignment of weights to decision trees in the RF. This approach is implemented in various algorithms [29–34]. However, most of these algorithms have a disadvantage. The weights are assigned to trees independently of training or testing examples, i.e., each weight characterizes trees on average, over all training examples, and it does not take into account any feature vector. Moreover, the weights do not have training parameters which usually make the model more flexible and accurate.

Contamination model in attention mechanisms. There are several models, which use imprecise probabilities in order to model the lack of sufficient training data. One of the first models is the so-called Credal Decision Tree, which is based on applying the imprecise probability theory to classification and proposed in [35]. Following this work, a number of models, based on imprecise probabilities, were presented in [36–39], where the imprecise Dirichlet model is used. This model can be regarded as a reparametrization of the

imprecise ϵ-contamination model, which is applied to LARF. The imprecise ϵ-contamination model has been also applied to machine learning methods, for example, to the support vector machine [40] or to the RF [41]. The attention-based RF applying the imprecise ϵ-contamination model to the parametric attention mechanism was proposed in [8,15]. However, there are no other works which use the imprecise models in order to implement the attention mechanism.

3. Nadaraya–Watson Regression and the Attention Mechanism

The basis of the attention mechanism can be considered in the framework of the Nadaraya–Watson kernel regression model [12,13], which estimates a function f as a locally weighted average using a kernel as a weighting function. Suppose the dataset is represented by n examples $(\mathbf{x}_1, y_1), ..., (\mathbf{x}_n, y_n)$, where $\mathbf{x}_i = (x_{i1}, ..., x_{im}) \in \mathbb{R}^m$ is a feature vector consisting of m features; $y_i \in \mathbb{R}$ is a regression output. The regression task is to construct a regressor $f : \mathbb{R}^m \to \mathbb{R}$, which can predict the output value $\tilde{y}$ of a new observation $\mathbf{x}$, using the dataset.

The Nadaraya–Watson kernel regression estimates the regression output $\tilde{y}$ corresponding to a new input feature vector $\mathbf{x}$, as follows [12,13]:

$$\tilde{y} = \sum_{i=1}^{n} \alpha(\mathbf{x}, \mathbf{x}_i) y_i, \tag{1}$$

where weight $\alpha(\mathbf{x}, \mathbf{x}_i)$ conforms with a relevance of the feature vector $\mathbf{x}_i$ to the vector $\mathbf{x}$.

It can be observed from the above that the Nadaraya–Watson regression model estimates $\tilde{y}$ as a weighted sum of training outputs y_i from the dataset so that their weights depend on the location of $\mathbf{x}_i$ relative to $\mathbf{x}$. This means that the closer $\mathbf{x}_i$ is to $\mathbf{x}$, the greater weight is assigned to y_i.

According to the Nadaraya–Watson kernel regression [12,13], weights can be defined by means of the kernel K as a function of the distance between the vectors $\mathbf{x}_i$ and $\mathbf{x}$. The kernel estimates how $\mathbf{x}_i$ is close to $\mathbf{x}$. Then, the weight is written as follows:

$$\alpha(\mathbf{x}, \mathbf{x}_i) = \frac{K(\mathbf{x}, \mathbf{x}_i)}{\sum_{j=1}^{n} K(\mathbf{x}, \mathbf{x}_j)}. \tag{2}$$

One of the popular kernels is the Gaussian kernel. It produces weights of the form:

$$\alpha(\mathbf{x}, \mathbf{x}_i) = \sigma\left(-\frac{\|\mathbf{x} - \mathbf{x}_i\|^2}{\tau}\right), \tag{3}$$

where τ is a tuning (temperature) parameter; $\sigma(\cdot)$ is a notation of the softmax operation.

In terms of the attention mechanism [42], the vector $\mathbf{x}$, vectors $\mathbf{x}_i$, outputs y_i, and weight $\alpha(\mathbf{x}, \mathbf{x}_i)$ are called the query, keys, values, and the attention weight, respectively. Weights $\alpha(\mathbf{x}, \mathbf{x}_i)$ can be extended by incorporating trainable parameters. In particular, parameter τ can also be regarded as the trainable parameter.

Many forms of parametric attention weights, which also define the attention mechanisms, have been proposed, e.g., the additive attention [42], the multiplicative or dot-product attention [16,43]. We also consider the attention weights based on the Gaussian kernels, i.e., producing the softmax operation. However, the parametric forms of the attention weights will be quite different from many popular attention operations.

4. Two-Level Attention-Based Random Forest

One of the powerful machine learning models handling tabular data is the RF, which can be regarded as an ensemble of T decision trees so that each tree is trained on a subset of examples randomly selected from the training set. In the original RF, the final RF prediction $\tilde{y}$ for the testing example $\mathbf{x}$ is determined by averaging predictions $\tilde{y}_1, ..., \tilde{y}_T$ obtained for all trees.

Let $\mathcal{J}_k(\mathbf{x})$ be the index set of examples which fall into the same leaf in the k-th tree as $\mathbf{x}$. One of the ways to construct the attention-based RF is to introduce the mean vector $\mathbf{A}_k(\mathbf{x})$ defined as the mean of the training vectors $\mathbf{x}_j$ which fall into the same leaf as $\mathbf{x}$. However, this simple definition can be extended by incorporating the Nadaraya–Watson regression into the leaf. In this case, we can write

$$\mathbf{A}_k(\mathbf{x}) = \sum_{j \in \mathcal{J}_k(\mathbf{x})} \mu(\mathbf{x}, \mathbf{x}_j)\mathbf{x}_j, \tag{4}$$

where $\mu(\mathbf{x}, \mathbf{x}_j)$ is the attention weight in accordance with the Nadaraya–Watson kernel regression.

In fact, (4) can be regarded as the self-attention. The idea behind (4) is that we find the mean value of $\mathbf{x}$ by assigning weights to training examples which fall into the corresponding leaf in accordance with their vicinity to the vector $\mathbf{x}$.

In the same way, we can define the mean value of regression outputs corresponding to examples falling into the same leaf as $\mathbf{x}$:

$$B_k(\mathbf{x}) = \sum_{j \in \mathcal{J}_k(\mathbf{x})} \mu(\mathbf{x}, \mathbf{x}_j)y_j. \tag{5}$$

Expression (5) can be regarded as the attention. The idea behind (5) is to obtain the prediction provided by the corresponding leaf by using the standard attention mechanism or Nadaraya–Watson regression. In other words, we weigh predictions provided by the k-th leaf of a tree in accordance with the distance between the feature vector $\mathbf{x}$, which falls into the k-th leaf, and all the feature vectors $\mathbf{x}_j$ which fall into the same leaf. It should be noted that the original regression tree provides the averaged prediction; i.e., it corresponds to the case when all $\mu(\mathbf{x}, \mathbf{x}_j)$ are identical for all $j \in \mathcal{J}_k(\mathbf{x})$ and equal to $1/\#\mathcal{J}_k(\mathbf{x})$.

We suppose that the attention mechanisms used above are non-parametric. This implies that weights do not have trainable parameters. It is assumed that

$$\sum_{j \in \mathcal{J}_k(\mathbf{x})} \mu(\mathbf{x}, \mathbf{x}_j) = 1. \tag{6}$$

The "leaf" attention introduced above can be regarded as the first-level attention in a hierarchy of the attention mechanisms. It characterizes how the feature vector $\mathbf{x}$ fits the corresponding tree.

If we suppose that the whole RF consists of T decision trees, then, the set of $\mathbf{A}_k(\mathbf{x})$, $k = 1, ..., T$, in the framework of the attention mechanism, can be regarded as a set of keys for every $\mathbf{x}$, the set of $B_k(\mathbf{x})$, $k = 1, ..., T$, can be regarded as a set of values. This implies that the final prediction $\tilde{y}$ of the RF can be computed by using the Nadaraya–Watson regression, namely,

$$\tilde{y} = f(\mathbf{x}, \mathbf{w}) = \sum_{k=1}^{T} \alpha(\mathbf{x}, \mathbf{A}_k(\mathbf{x}), \mathbf{w})B_k(\mathbf{x}). \tag{7}$$

Here, $\alpha(\mathbf{x}, \mathbf{A}_k(\mathbf{x}), \mathbf{w})$ is the attention weight with the vector $\mathbf{w} = (w_1, ..., w_T)$ of trainable parameters which belong to a set $\mathcal{W}$ so that they are assigned to each tree. The attention weight α is defined by the distance between $\mathbf{x}$ and $\mathbf{A}_k(\mathbf{x})$. It is assumed due to properties of the attention weights in the Nadaraya–Watson regression that the following condition is valid:

$$\sum_{k=1}^{T} \alpha(\mathbf{x}, \mathbf{A}_k(\mathbf{x}), \mathbf{w}) = 1. \tag{8}$$

The above "random forest" attention can be regarded as the second-level attention which assigns weights to trees in accordance with their impact on the RF prediction corresponding to $\mathbf{x}$.

The main idea behind the approach is to use the above attention mechanisms jointly. After substituting (4) and (5) into (7), we obtain

$$\tilde{y}(\mathbf{x}) = f(\mathbf{x}, \mathbf{w}) = \sum_{k=1}^{T} \alpha(\mathbf{x}, \mathbf{A}_k(\mathbf{x}), \mathbf{w}) \sum_{j \in \mathcal{J}_k(\mathbf{x})} y_j \mu(\mathbf{x}, \mathbf{x}_j), \tag{9}$$

or

$$\tilde{y}(\mathbf{x}) = \sum_{k=1}^{T} \alpha \left(\mathbf{x}, \sum_{i \in \mathcal{J}_k(\mathbf{x})} \mu(\mathbf{x}, \mathbf{x}_i) \mathbf{x}_i, \mathbf{w} \right) \sum_{j \in \mathcal{J}_k(\mathbf{x})} y_j \mu(\mathbf{x}, \mathbf{x}_j). \tag{10}$$

A scheme of the two-level attention is shown in Figure 1. It is observed from Figure 1 how the attention at the second level depends on the "leaf" attention at the first level.

In total, we obtain the trainable attention-based RF with parameters $\mathbf{w}$, which are defined by minimizing the expected loss function over the set $\mathcal{W}$ of parameters, respectively, as follows:

$$\mathbf{w}_{opt} = \arg \min_{\mathbf{w} \in \mathcal{W}} \sum_{s=1}^{n} L(\tilde{y}(\mathbf{x}_s), y_s, \mathbf{w}). \tag{11}$$

The loss function can be rewritten as the following:

$$\begin{aligned} &\sum_{s=1}^{n} L(\tilde{y}(\mathbf{x}_s), y_s, \mathbf{w}) = \sum_{s=1}^{n} (y_s - \tilde{y}(\mathbf{x}_s))^2 \\ &= \sum_{s=1}^{n} \left(y_s - \sum_{k=1}^{T} \sum_{j \in \mathcal{J}_k(\mathbf{x}_s)} y_j \mu(\mathbf{x}_s, \mathbf{x}_j) \cdot \alpha \left(\mathbf{x}_s, \sum_{i \in \mathcal{J}_k(\mathbf{x}_s)} \mu(\mathbf{x}_s, \mathbf{x}_i) \mathbf{x}_i, \mathbf{w} \right) \right)^2. \end{aligned} \tag{12}$$

The optimal trainable parameters $\mathbf{w}$ are computed depending on forms of the attention weights α in the optimization problem (12). It should be noted that the problem (12) may be complex from the computation point of view. Therefore, one of our results is to propose such a form of the attention weights α that reduces the problem (12) to a convex quadratic optimization problem, whose solution does not meet any difficulties.

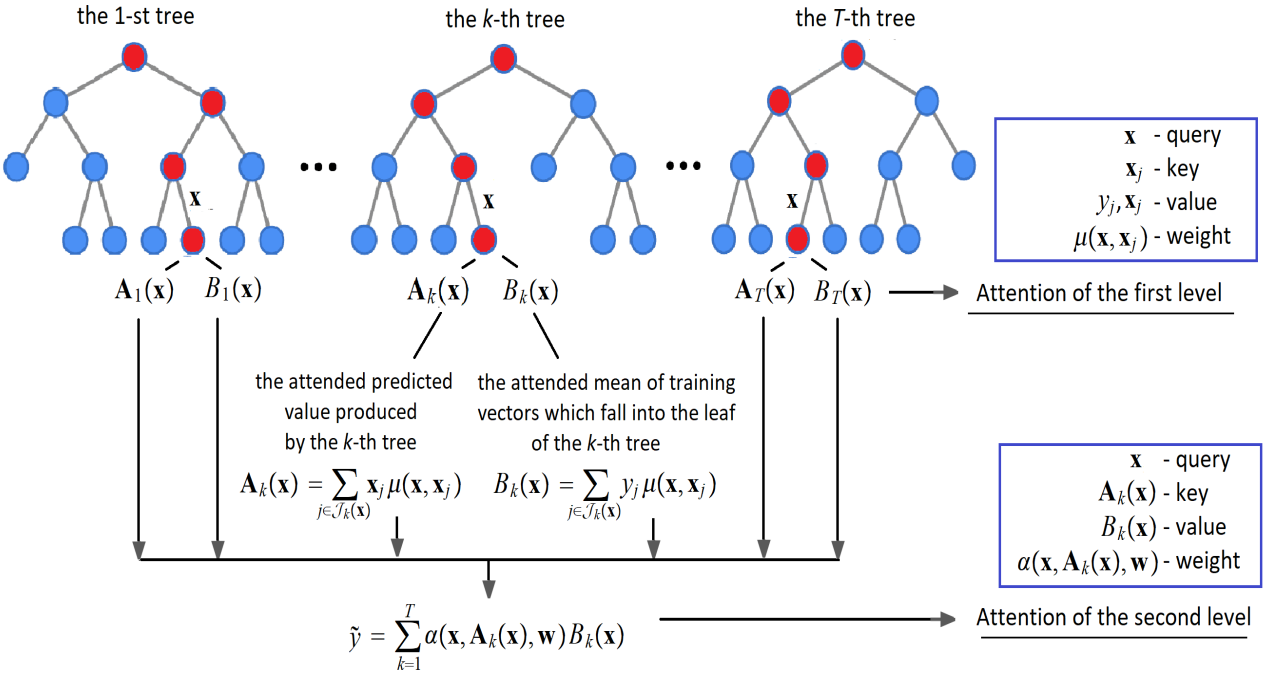

Figure 1. A scheme of the proposed two-level hierarchical attention model applied to the RF.

It is important to point out that the additional sets of trainable parameters can be introduced into the definition of the attention weights $\mu(\mathbf{x}, \mathbf{x}_j)$. On the one hand, we obtain more flexible attention mechanisms in this case due to the parametrization of the training weights $\mu(\mathbf{x}, \mathbf{x}_j)$. On the other hand, many trainable parameters lead to the increasing complexity of the optimization problem (11) and the possible overfitting of the whole RF.

5. Modifications of the Two-Level Attention-Based Random Forest

Different configurations of LARF produce a set of models which depend on trainable parameters of the two-level attention and its implementation. A classification of models and their notations are shown in Table 1. In order to explain the classification, two subsets of the attention parameters should be considered:

1. Parameters produced by the contamination probability distributions of Huber's ϵ-contamination model in the form of the vector $\mathbf{w}$, whose length coincides with the number of trees.
2. Parameters $\epsilon_1, ..., \epsilon_M$ of contamination in the mixture of M in the Huber's contamination models, which define the imprecision of the mixture model.

The following models can be constructed depending on trainable parameters and on using the "leaf" attention, i.e., the two-level attention mechanism:

- ϵ-ARF: The attention-based forest with learning ϵ as an attention parameter, but without the training vector $\mathbf{w}$, i.e., $w_k = 1/T$, $k = 1, ..., T$, and without the "leaf" attention;
- w-ARF: The attention-based forest with the learning vector $\mathbf{w}$ as attention parameters and without the "leaf" attention;
- ϵ-LARF: The attention-based forest with learning ϵ as an attention parameter, but without the training vector $\mathbf{w}$ and with the "leaf" attention, i.e., by using the two-level attention mechanism;
- w-LARF: The attention-based forest with the learning vector $\mathbf{w}$ as attention parameters and with the "leaf" attention, i.e., by using the two-level attention mechanism;
- ϵ-w-ARF: The attention-based forest with the learning vector $\mathbf{w}$ and the parameter ϵ as attention parameters and without the "leaf" attention;
- ϵ-w-LARF: The attention-based forest with the learning vector $\mathbf{w}$ and the parameter ϵ as attention parameters and with the "leaf" attention;
- ϵM-ARF: The attention-based forest with the learning parameters $\epsilon_1, ..., \epsilon_M$ as attention parameters with $w_k = 1/T$, $k = 1, ..., T$, and without the "leaf" attention;
- ϵM-LARF: The attention-based forest with the learning parameters $\epsilon_1, ..., \epsilon_M$ as attention parameters with $w_k = 1/T$, $k = 1, ..., T$, and with the "leaf" attention;
- ϵM-w-ARF: The attention-based forest with the learning vector $\mathbf{w}$ and the parameters $\epsilon_1, ..., \epsilon_M$ as attention parameters and without the "leaf" attention;
- ϵM-w-LARF: The attention-based forest with the learning vector $\mathbf{w}$ and the parameters $\epsilon_1, ..., \epsilon_M$ as attention parameters and with the "leaf" attention.

Models ϵ-ARF, ϵ-LARF, ϵ-$\mathbf{w}$-ARF, and ϵ-$\mathbf{w}$-LARF are not presented in Table 1 because they are special cases of models ϵM-ARF, ϵM-LARF, ϵM-$\mathbf{w}$-ARF, and ϵM-$\mathbf{w}$-LARF, respectively, by $M = 1$.

Table 1. Classification of the attention-based RF models proposed and studied in this paper.

	Tuning ϵ		**Trainable $\epsilon_1, ..., \epsilon_M$**	
	Fixed w	**Trainable w**	**Fixed w**	**Trainable w**
Without the "leaf" attention	-	**w**-ARF	ϵM-ARF	ϵM-**w**-ARF
With the "leaf" attention	-	**w**-LARF	ϵM-LARF	ϵM-**w**-LARF

5.1. Huber's Contamination Model and the Basic Two-Level Attention

In order to simplify the optimization problem (12) and to effectively solve it, we offer to represent the attention weights $\alpha(\mathbf{x}, \mathbf{A}_k(\mathbf{x}), \mathbf{w})$ by using Huber's ϵ-contamination model [14]. The idea to represent the attention weight by means of the ϵ-contamination model has been proposed in [8]. We use this idea to incorporate the ϵ-contamination model into the optimization problem (12) and to construct the first modifications of LARF.

Let us provide a brief introduction to Huber's ϵ-contamination model. The model considers a set of probability distributions of the form $F(\mathbf{x}) = (1-\epsilon) \cdot P(\mathbf{x}) + \epsilon \cdot R$. Here, $P(\mathbf{x}) = (p_1(\mathbf{x}), ..., p_T(\mathbf{x}))$ is a discrete probability distribution contaminated by another probability distribution, denoted as $R = (r_1, ..., r_T)$, which can be arbitrary in the unit simplex having the dimension T. It is important to note that the distribution P depends on the feature vector $\mathbf{x}$, i.e., it is different for every vector $\mathbf{x}$, whereas the distribution R does not depend on $\mathbf{x}$. The contamination parameter $\epsilon \in [0, 1]$ controls the impact of the contamination probability distribution R on the distribution $P(\mathbf{x})$. Since the distribution R can be arbitrary, the set of the distributions F forms a subset of the unit simplex so that its size depends on the parameter ϵ. If $\epsilon = 0$, then the subset of distributions F is reduced to the single distribution $P(\mathbf{x})$. In case $\epsilon = 1$, the set of $F(\mathbf{x})$ is the whole unit simplex.

Following the common definition of the attention weights through the softmax operation with the parameter τ, we propose to define each probability in $P(\mathbf{x})$ as

$$p_k(\mathbf{x}) = \sigma\Big(-\|\mathbf{x} - \mathbf{A}_k(\mathbf{x})\|^2 / \tau\Big).$$

This implies that the distribution $P(\mathbf{x})$ characterizes how the feature vector $\mathbf{x}$ is far from the vector $\mathbf{A}_k(\mathbf{x})$ in all trees of the RF. Let us suppose that the probability distribution R is the vector of the trainable parameters $\mathbf{w}$. The idea is to train the parameters $\mathbf{w}$ to achieve the best accuracy of the RF. After substituting the softmax operation into the attention weight α, we obtain:

$$\alpha(\mathbf{x}, \mathbf{A}_k(\mathbf{x}), \mathbf{w}) = (1-\epsilon) \cdot \sigma\Big(-\|\mathbf{x} - \mathbf{A}_k(\mathbf{x})\|^2 / \tau\Big) + \epsilon \cdot w_k, \; k = 1, ..., T. \tag{13}$$

One can observe from (13) that the attention weight is linearly dependent on the trainable parameters $\mathbf{w} = (w_1, ..., w_T)$. It is important to note that the attention weight assigned to the k-th tree depends only on the k-th parameter w_k, but not on other elements of the vector $\mathbf{w}$. The parameter ϵ is a tuning parameter determined by means of the standard validation procedure. It should be noted that elements of the vector $\mathbf{w}$ are probabilities. Hence, we can write

$$\sum_{k=1}^{T} w_k = 1, \; w_k \geq 0, \; k = 1, ..., T. \tag{14}$$

This implies that the set $\mathcal{W}$ is the unit simplex of the dimension T.

Let us return to the attention weight $\mu(\mathbf{x}, \mathbf{x}_j)$ of the first level. The attention is nonparametric at the first level; therefore, the attention weight can be defined in the standard way by using the Gaussian kernel or the softmax operation with the parameter τ_0; i.e., it can be expressed in this form:

$$\mu(\mathbf{x}, \mathbf{x}_j) = \sigma\Big(-\|\mathbf{x} - \mathbf{x}_j\|^2 / \tau_0\Big). \tag{15}$$

Finally, we can rewrite the loss function (12) by taking into account the above definitions of the attention weights, as follows:

$$\min_{\mathbf{w}\in\mathcal{W}} \sum_{s=1}^{n} L(\tilde{y}(\mathbf{x}_s), y_s, \mathbf{w})$$
$$= \min_{\mathbf{w}\in\mathcal{W}} \sum_{s=1}^{n} \left(y_s - \sum_{k=1}^{T} ((1-\epsilon)C_k(\mathbf{x}_s) - \epsilon D_k(\mathbf{x}_s) w_k) \right)^2, \tag{16}$$

where

$$C_k(\mathbf{x}_s) = \sum\nolimits_{l\in\mathcal{J}_k(\mathbf{x}_s)} y_l \cdot \sigma\left(-\frac{\|\mathbf{x}_s - \mathbf{x}_l\|^2}{\tau_0} \right)$$
$$\times \sigma\left(-\frac{\left\| \mathbf{x} - \sum_{i\in\mathcal{J}_k(\mathbf{x}_s)} \mathbf{x}_i \cdot \sigma\left(-\|\mathbf{x}_s - \mathbf{x}_i\|^2 / \tau_0 \right) \right\|^2}{\tau} \right), \tag{17}$$

$$D_k(\mathbf{x}_s) = \sum\nolimits_{j\in\mathcal{J}_k(\mathbf{x}_s)} y_j \cdot \sigma\left(-\frac{\|\mathbf{x}_s - \mathbf{x}_j\|^2}{\tau_0} \right). \tag{18}$$

One can observe from the above that $C_k(\mathbf{x}_s)$ and $D_k(\mathbf{x}_s)$ do not depend on the parameters $\mathbf{w}$. Therefore, the objective function (16) jointly with the simple constraints $\mathbf{w} \in \mathcal{W}$ or (14) is the standard quadratic optimization problem, which can be solved by means of many available efficient algorithms. The corresponding model is called **w-**LARF. The notation means that trainable parameters are $\mathbf{w}$. The same model without "leaf" attention is denoted as **w-**ARF. It coincides with the model ϵ-ABRF proposed in [8].

It should be worth knowing that the problem (16) is similar to the optimization problem stated in [7,8]. However, it turns out that the addition of the "leaf" attention significantly improves the RF, as it will be demonstrated by many numerical experiments with real data.

5.2. Models with the Trainable Contamination Parameter ϵ

One of the important contributions to the work, which makes the proposed model different from the model presented in [7,8], is the idea of learning the contamination parameter ϵ jointly with the parameters $\mathbf{w}$. However, this idea leads to a complex optimization problem, where gradient-based algorithms have to be used. In order to avoid using these algorithms and to tackle a simple optimization problem, we consider two ways. The first way is just to assign the same value $1/T$ to all parameters w_k. Then, the optimization problem (16) can be rewritten as the following:

$$\min_{0\le\epsilon\le 1} \sum_{s=1}^{n} \left(y_s - \sum_{k=1}^{T} \left((1-\epsilon)C_k(\mathbf{x}_s) - \epsilon D_k(\mathbf{x}_s) \frac{1}{T} \right) \right)^2. \tag{19}$$

We have a simple quadratic optimization problem with one variable ϵ. Let us call the corresponding model as ϵ-LARF. The notation means that the trainable parameter is ϵ. The same model without the "leaf" attention is denoted as ϵ-ARF.

Another way is to introduce new variables $\gamma_k = \epsilon w_k$, $k = 1, ..., T$. Then, the optimization problem (16) can be rewritten in this form:

$$\min_{\gamma_1,...,\gamma_T,\epsilon} \sum_{s=1}^{n} \left(y_s - (1-\epsilon) \sum_{k=1}^{T} C_k(\mathbf{x}_s) - \sum_{k=1}^{T} \gamma_k D_k(\mathbf{x}_s) \right)^2, \tag{20}$$

subject to

$$\sum_{k=1}^{T} \gamma_k = \epsilon,\ \gamma_k \ge 0,\ k = 1, ..., T, \tag{21}$$

$$0 + \varsigma \leq \epsilon \leq 1. \tag{22}$$

We again deal with the quadratic optimization problem with new optimization variables $\gamma_1, ..., \gamma_T, \epsilon$ and linear constraints. The parameter ς takes a small value to avoid the case $\epsilon = 0$. The corresponding model is denoted as ϵ-**w**-LARF. The notation means that the trainable parameters are **w** and ϵ. The same model without the "leaf" attention is denoted as ϵ-**w**-ARF.

5.3. Mixture of Contamination Models

Another important contribution is an attempt to search for an optimal value of the temperature parameter τ in (13) or in (17). We propose an approximate approach, which can significantly improve the model. Let us introduce a finite set $\{\tau_1, ..., \tau_M\}$ of M values of the parameter τ. Here, M can be regarded as a tuning integer parameter which impacts the number of all training parameters. Large values of M may lead to a large number of training parameters and the corresponding overfitting. Small values of M may lead to an inexact approximation of τ.

Before considering how the optimization problem can be rewritten with allowance for the above information, we again return to the attention weight α in (13) and represent it as follows:

$$\alpha(\mathbf{x}, \mathbf{A}_k(\mathbf{x}), w_k) = \frac{1}{M} \sum_{j=1}^{M} \alpha_j(\mathbf{x}, \mathbf{A}_k(\mathbf{x}), w_k), \tag{23}$$

where

$$\alpha_j(\mathbf{x}, \mathbf{A}_k(\mathbf{x}), w_k) = (1 - \epsilon_j)\sigma\left(-\frac{\|\mathbf{x} - \mathbf{A}_k(\mathbf{x})\|^2}{\tau_j}\right) + \epsilon_j w_k. \tag{24}$$

We have a mixture of M contamination models with the different contamination parameters ϵ_j. It is obvious that the sum of new weights α over $k = 1, ..., T$ is 1 because the sum of each $\alpha_j(\mathbf{x}, \mathbf{A}_k(\mathbf{x}), w_k)$ over $k = 1, ..., T$ is also 1. Each $\alpha_j(\mathbf{x}, \mathbf{A}_k(\mathbf{x}), w_k)$ forms a small simplex so that its center is defined by τ_j and its size is defined by τ_j. The corresponding sets of possible attention weights are depicted in Figure 2, where the unit simplex by $T = 3$ includes small simplices corresponding to three ($M = 3$) contamination models $\alpha_j(\mathbf{x}, \mathbf{A}_k(\mathbf{x}), w_k)$, $j = 1, 2, 3$, with different centers and different contamination parameters $\epsilon_1, \epsilon_2, \epsilon_3$. The "mean" simplex of weights $\alpha(\mathbf{x}, \mathbf{A}_k(\mathbf{x}), w_k)$ is depicted by using dashed sides. Parameters **w** are optimized so that the attention weights will be located in the "mean" simplex.

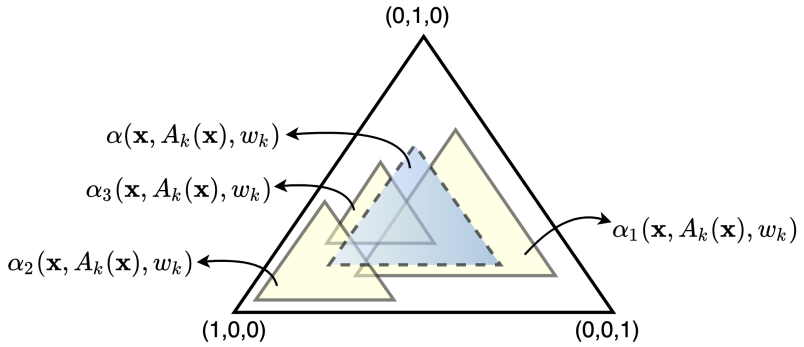

Figure 2. The unit simplex of possible attention weights, which includes small simplices corresponding to three contamination models with different centers and different contamination parameters ϵ and the "mean" simplex (with dashed sides), which defines the set of the final attention weights.

Let us prove that the resulting "mean" model represents the ϵ-contamination model with the contamination parameter ϵ. We use P_j to denote the probability distibution $P_j = (p_j^{(1)}, ..., p_j^{(T)})$ as follows:

$$P_j = \sigma\left(-\frac{\|\mathbf{x} - \mathbf{A}_k(\mathbf{x})\|^2}{\tau_j}\right). \tag{25}$$

Then, we can write

$$\begin{aligned} \alpha(\mathbf{x}, \mathbf{A}_k(\mathbf{x}), w_k) &= \frac{1}{M}\sum_{j=1}^{M}(1-\epsilon_j)P_j + \frac{1}{M}\sum_{j=1}^{M}\epsilon_j w_k \\ &= \frac{1}{M}\sum_{j=1}^{M}(1-\epsilon_j)P_j + \epsilon w_k, \end{aligned} \tag{26}$$

where

$$\epsilon = \frac{1}{M}\sum_{j=1}^{M}\epsilon_j. \tag{27}$$

Suppose there is a probability distribution $Q = (q^{(1)}, ..., q^{(T)})$; therefore, it holds

$$\frac{1}{M}\sum_{j=1}^{M}(1-\epsilon_j)P_j = (1-\epsilon)Q. \tag{28}$$

If we prove that the probability distribution Q exists, the resulting "mean" model is the ϵ-contamination model. Let us find sums of the left and the right sides of (28) over $i = 1, ..., T$. Hence, we obtain

$$\frac{1}{M}\sum_{j=1}^{M}(1-\epsilon_j)\sum_{i=1}^{T}p_j^{(i)} = (1-\epsilon)\sum_{i=1}^{T}q^{(i)}. \tag{29}$$

Substituting (27) into (29), we obtain

$$\sum_{i=1}^{T}q^{(i)} = 1,$$

as it was to be proved.

The introduced mixture of the contamination models can be regarded as a multi-head attention to some extent, where every "head" is produced by using a certain parameter τ_j.

Let us represent the softmax operation in (13) jointly with the factor $(1-\epsilon)$ as follows:

$$(1-\epsilon)\cdot\sigma\left(\frac{-\|\mathbf{x} - \mathbf{A}_k(\mathbf{x})\|^2}{\tau}\right) = \frac{1}{M}\sum_{j=1}^{M}(1-\epsilon_j)\cdot\sigma\left(\frac{-\|\mathbf{x} - \mathbf{A}_k(\mathbf{x})\|^2}{\tau_j}\right). \tag{30}$$

It can be observed from (30) that new parameters $\epsilon_1, ..., \epsilon_M$ along with $\tau_1, ..., \tau_M$ are introduced in the place of ϵ and τ, respectively. Term $(1-\epsilon)\cdot C_k(\mathbf{x}_s)$ in (16) and (17) is replaced with the following terms:

$$\frac{1}{M}\sum_{j=1}^{M}(1-\epsilon_j)C_k^{(j)}(\mathbf{x}_s), \tag{31}$$

where

$$C_k^{(j)}(\mathbf{x}_s) = \sum_{l \in \mathcal{J}_k(\mathbf{x}_s)} y_l \cdot \sigma\left(-\frac{\|\mathbf{x}_s - \mathbf{x}_l\|^2}{\tau_0}\right) \times \sigma\left(-\frac{\left\|\mathbf{x}_s - \sum_{i \in \mathcal{J}_k(\mathbf{x}_s)} \mathbf{x}_i \mu(\mathbf{x}_s, \mathbf{x}_i)\right\|^2}{\tau_j}\right). \quad (32)$$

Finally, we observed the following optimization problem:

$$\min_{\gamma_1,...,\gamma_T,\epsilon} \sum_{s=1}^{n} \left(y_s - \frac{1}{M} \sum_{j=1}^{M} (1-\epsilon_j) \sum_{k=1}^{T} C_k^{(j)}(\mathbf{x}_s) - \frac{1}{M} \sum_{j=1}^{M} \epsilon_j \sum_{k=1}^{T} w_k D_k(\mathbf{x}_s) \right)^2, \quad (33)$$

subject to

$$\sum_{k=1}^{T} w_k = 1,\ w_k \geq 0,\ k = 1, ..., T.$$

Let us introduce new variables $\gamma_k^{(j)} = w_k \epsilon_j$, $k = 1, ..., T$, $j = 1, ..., M$. Hence, we can write the optimization problem with the new $M \cdot T + M$ variables as

$$\min_{\gamma_1,...,\gamma_T,\epsilon} \sum_{s=1}^{n} \left(y_s - \frac{1}{M} \sum_{j=1}^{M} (1-\epsilon_j) \sum_{k=1}^{T} C_k^{(j)}(\mathbf{x}_s) - \frac{1}{M} \sum_{j=1}^{M} \sum_{k=1}^{T} \gamma_k^{(j)} D_k(\mathbf{x}_s) \right)^2, \quad (34)$$

subject to

$$\sum_{k=1}^{T} \gamma_k^{(j)} = \epsilon_j,\ \gamma_k^{(j)} \geq 0,\ k = 1, ..., T,\ , j = 1, ..., M, \quad (35)$$

$$0 + \varsigma \leq \epsilon_j \leq 1. \quad (36)$$

We again face the quadratic optimization problem with linear constraints. The corresponding model will be denoted as ϵM-**w**-ARF or ϵM-**w**-LARF, depending on applying the "leaf" attention. The notation means that the trainable parameters are **w** and $\epsilon_1, ..., \epsilon_M$. Additionally, we will use the same model, but with condition $w_k = 1/T$ for all $k = 1, ..., T$. The corresponding models are denoted as ϵM-ARF or ϵM-LARF.

6. Numerical Experiments

Let us introduce notations for different models of the attention-based RFs.

1. RF (ERT): the original RF (the ERT) without applying attention mechanisms;
2. ARF (LARF): the attention-based forest which has the following modifications: ϵM-ARF, ϵM-LARF, ϵM-w-ARF, and ϵM-w-LARF.

In all experiments, RFs as well as ERTs consist of 100 trees. To select the best tuning parameters in numerical experiments, a 3-fold cross-validation on the training set consisting of $n_{\text{tr}} = 4n/5$ examples with 100 repetitions is performed. The search for the best parameter τ_0 is carried out by considering all its values in a predefined grid. A cross-validation procedure is subsequently used to select their appropriate values. The testing set for computing the accuracy measures is comprised of $n_{\text{test}} = n/5$ examples. In order to obtain desirable estimates of the vectors $\mathbf{A}_k(\mathbf{x})$ and $B_k(\mathbf{x})$, all trees in the experiments are trained

so that at least 10 examples fall into every leaf of a tree. Value 10 for the number of examples is taken for two reasons. On the one hand, we have to compute the mean vectors $\mathbf{A}_k(\mathbf{x})$ and $B_k(\mathbf{x})$ and to obtain unbiased estimators. On the other hand, it is difficult to expect that a large number of examples will fall into every leaf by a small number of training examples. Therefore, our prior experiments have demonstrated that this parameter should be 10. We also use all features at each split of decision trees.

We do not consider models ϵ-ARF, **w**-ARF, ϵ-LARF, and **w**-LARF, because they can be regarded as special cases of the corresponding models ϵM-ARF, ϵM-**w**-ARF, ϵM-LARF, and ϵM-**w**-LARF when $M = 1$. The value of M is an integer tuning parameter, and it is tuned in the interval from 1 to 20. Set $\{\tau_1, ..., \tau_M\}$ of the softmax operation parameters is defined as $\left\{10^{-\lfloor M/2 \rfloor}, 10^{-\lfloor M/2 \rfloor + 1}, ..., 10^{0}, ..., 10^{\lfloor M/2 \rfloor - 1}, 10^{\lfloor M/2 \rfloor}\right\}$. In particular, if $M = 1$, then the set of τ consists of one element $\tau = 1$. The parameter of the first-level attention in the "leaf" τ_0 is taken equal to 1.

Numerical results are presented in tables where the best results are shown in bold. The coefficient of determination denoted R^2 and the mean absolute error (MAE) are used for the regression evaluation. The greater value of the coefficient of determination and the smaller MAE we have, the better results we achieve.

The proposed approach is studied by applying datasets which are taken from open sources. The dataset Diabetes is downloaded from the R Packages; datasets Friedman 1, 2 and 3 are retrieved from the site: https://www.stat.berkeley.edu/~breiman/bagging.pdf (accessed on 20 April 2023); datasets Regression and Sparse are taken from package "Scikit-Learn"; datasets Wine Red, Boston Housing, Concrete, Yacht Hydrodynamics, Airfoil can be found in the UCI Machine Learning Repository [44]. These datasets with their numbers of features m and numbers of examples n are given in Table 2. A more detailed information can be found from the aforementioned data resources.

Table 2. A brief introduction about the regression data sets.

Data Set	Abbreviation	*m*	*n*
Diabetes	Diabetes	10	442
Friedman 1	Friedman 1	10	100
Friedman 2	Friedman 2	4	100
Friedman 3	Friedman 3	4	100
Scikit-Learn Regression	Regression	100	100
Scikit-Learn Sparse Uncorrelated	Sparse	10	100
UCI Wine red	Wine	11	1599
UCI Boston Housing	Boston	13	506
UCI Concrete	Concrete	8	1030
UCI Yacht Hydrodynamics	Yacht	6	308
UCI Airfoil	Airfoil	5	1503

Values of the measure R^2 for several models, including RF, ϵM-**w**-ARF, ϵM-**w**-LARF, ϵM-ARF, and ϵM-LARF, are shown in Table 3. The results are obtained by training the RF. The optimal values of τ_0 are also given in the table. It can be observed from Table 3 that ϵM-**w**-LARF outperforms all models for most datasets. Moreover, Table 3 shows that the two-level attention models (ϵM-**w**-LARF and ϵM-LARF) provide better results than models which do not use the "leaf" attention (ϵM-**w**-ARF and ϵM-ARF). It should be also noted that all attention-based models outperform the original RF. The same relationship between the models occurs for another accuracy measure (MAE). It is clearly shown in Table 4.

Table 3. Values of R^2 for comparison of models based on the RF.

Data Set	τ_0	RF	ϵM-w-ARF	ϵM-w-LARF	ϵM-ARF	ϵM-LARF
Diabetes	0.01	0.416	0.419	**0.434**	0.425	0.426
Friedman 1	1	0.459	0.470	**0.524**	0.438	0.472
Friedman 2	1	0.841	0.887	**0.933**	0.886	0.916
Friedman 3	1	0.625	0.708	**0.749**	0.675	0.704
Airfoil	100	0.823	0.844	0.914	0.822	**0.917**
Boston	1	0.814	0.820	**0.870**	0.819	0.856
Concrete	10	0.845	0.857	**0.896**	0.844	**0.896**
Wine	1	0.433	0.421	**0.481**	0.421	0.477
Yacht	0.1	0.981	0.989	**0.993**	0.981	0.982
Regression	0.1	0.380	0.434	**0.455**	0.361	0.409
Sparse	1	0.470	0.489	**0.641**	0.535	0.630

The best obtained results on each dataset are shown in bold.

Table 4. Values of MAE for comparison of models based on the RF.

Data Set	RF	ϵM-w-ARF	ϵM-w-LARF	ϵM-ARF	ϵM-LARF
Diabetes	44.92	44.95	**44.61**	44.79	44.81
Friedman 1	2.540	2.545	**2.411**	2.595	2.473
Friedman 2	111.7	95.29	**72.71**	92.44	74.24
Friedman 3	0.154	0.130	0.135	0.144	**0.129**
Airfoil	2.203	2.065	1.451	2.217	**1.416**
Boston	2.539	2.538	**2.148**	2.489	2.217
Concrete	4.834	4.676	**3.496**	4.883	3.615
Wine	0.451	0.459	**0.411**	0.461	0.417
Yacht	1.004	0.787	**0.611**	1.004	0.971
Regression	109.1	103.6	**101.3**	111.2	105.8
Sparse	1.908	1.871	**1.528**	1.772	1.543

The best obtained results on each dataset are shown in bold.

Another important question is how the attention-based models perform when the ERT is used. The corresponding values of R^2 and MAE are shown in Tables 5 and 6, respectively. Table 5 also contains the optimal values τ_0. In contrast to the case of using the RF, it can be observed from the tables that ϵM-LARF outperforms ϵM-**w**-LARF for several models. It can be explained by reducing the accuracy due to a larger number of training parameters (parameters **w**) and overfitting for small datasets. It is also worth noting that models based on ERTs provide better results than models based on RFs. However, this improvement is not significant. This is observed in Table 7, where the best results are collected for models based on ERTs and RFs. Table 7 shows that the results are identical for several datasets, namely, for datasets Friedman 1, 2, 3, Concrete, and Yacht. If one is to apply the t-test to compare the values of R^2 obtained for two models, then, according to [45], the t-statistics is distributed in accordance with the Student distribution with the $11 - 1$ degrees of freedom (11 datasets). The obtained p-value is $p = 0.071$. We can conclude that the outperformance of the ERT is not statistically significant because $p > 0.05$.

Table 5. Values of R^2 for comparison of models based on the ERT.

Data Set	τ_0	ERT	ϵM-w-ARF	ϵM-w-LARF	ϵM-ARF	ϵM-LARF
Diabetes	0.01	0.438	0.441	0.434	**0.471**	0.444
Friedman 1	1	0.471	0.471	**0.524**	0.441	0.495
Friedman 2	10	0.813	0.840	**0.933**	0.840	0.919
Friedman 3	1	0.570	0.569	**0.749**	0.569	0.637
Airfoil	100	0.802	0.804	**0.914**	0.804	0.909
Boston	10	0.831	0.834	0.870	0.834	**0.882**
Concrete	10	0.839	0.838	**0.896**	0.838	0.895
Wine	1	0.418	0.418	0.481	0.418	**0.486**
Yacht	1	0.988	0.988	**0.993**	0.988	**0.993**
Regression	0.1	0.402	0.429	0.455	0.429	**0.464**
Sparse	1	0.452	0.522	0.641	0.522	**0.663**

The best obtained results on each dataset are shown in bold.

Table 6. Values of MAE for comparison of models based on the ERT.

Data Set	ERT	w-ARF	w-LARF	ϵM-ARF	ϵM-LARF
Diabetes	44.549	44.271	44.614	44.271	**44.21**
Friedman 1	2.502	2.502	2.411	2.502	**2.388**
Friedman 2	123.0	113.7	72.71	113.7	**70.48**
Friedman 3	0.179	0.179	**0.135**	0.179	0.148
Airfoil	2.370	2.360	**1.451**	2.360	1.471
Boston	2.481	2.451	2.148	2.451	**2.023**
Concrete	5.119	5.124	**3.496**	5.124	3.659
Wine	0.464	0.464	**0.411**	0.464	0.412
Yacht	0.824	0.822	**0.611**	0.822	0.612
Regression	106.3	103.1	101.3	103.1	**100.0**
Sparse	1.994	1.820	1.528	1.820	**1.519**

The best obtained results on each dataset are shown in bold.

Table 7. Comparison of the best results provided by models based on RFs and ERTs.

Data Set	RF	ERT
Diabetes	0.434	**0.471**
Friedman 1	**0.524**	**0.524**
Friedman 2	**0.933**	**0.933**
Friedman 3	**0.749**	**0.749**
Airfoil	**0.917**	0.914
Boston	0.870	**0.882**
Concrete	**0.896**	**0.896**
Wine	0.481	**0.486**
Yacht	**0.993**	**0.993**
Regression	0.455	**0.464**
Sparse	0.641	**0.663**

The best obtained results on each dataset are shown in bold.

We take the number of trees in RFs equal to 100, because our goal is to compare RFs and the proposed modifications of LARF with the same parameters of numerical experiments. We also study how values of R^2 depend on different numbers of decision trees for the considered datasets. The corresponding numerical results for the RF and the ERT are shown in Tables 8 and 9, respectively, by 100, 400, 700, and 1000 trees. It can be observed in Tables 8 and 9 that R^2 insignificantly increases with the number of trees. However, it is important to point out that the largest values of R^2 for RFs and ERTs obtained by $T = 1000$ do not exceed the values of R^2 for LARF modifications presented in Tables 3–6.

Table 8. Values of R^2 for comparison of RFs by different numbers of decision trees.

Data Set	Numbers of Trees			
	100	**400**	**700**	**1000**
Diabetes	0.416	0.418	0.418	0.419
Friedman 1	0.459	0.465	0.465	0.465
Friedman 2	0.841	0.836	0.838	0.839
Friedman 3	0.625	0.625	0.626	0.627
Airfoil	0.823	0.824	0.824	0.824
Boston	0.814	0.815	0.816	0.816
Concrete	0.845	0.847	0.847	0.847
Wine	0.433	0.434	0.434	0.434
Yacht	0.981	0.982	0.982	0.982
Regression	0.380	0.397	0.398	0.399
Sparse	0.470	0.470	0.471	0.473

Table 9. Values of R^2 for comparison of ERTs by different numbers of decision trees.

Data Set	Numbers of Trees			
	100	**400**	**700**	**1000**
Diabetes	0.438	0.439	0.439	0.439
Friedman 1	0.471	0.475	0.474	0.475
Friedman 2	0.813	0.813	0.814	0.815
Friedman 3	0.570	0.565	0.567	0.568
Airfoil	0.802	0.803	0.803	0.803
Boston	0.831	0.831	0.832	0.833
Concrete	0.839	0.840	0.840	0.840
Wine	0.418	0.418	0.418	0.418
Yacht	0.988	0.988	0.988	0.988
Regression	0.402	0.390	0.387	0.390
Sparse	0.452	0.455	0.455	0.456

It should be pointed out that the proposed models can be regarded as extensions of the attention-based RF (ϵ-ABRF) presented in [8]. Therefore, it is also worth comparing the two-level attention models with ϵ-ABRF. Table 10 shows the values of R^2 obtained by using ϵ-ABRF and the best values of the proposed models when the RF and the ERT are used.

If we compare the results presented in Table 10 by applying the t-tests in accordance with [45], then tests for the proposed models and ϵ-ABRF based on the RF and the ERT provide p-values equal to $p = 0.00067$ and $p = 0.00029$, respectively. The tests demonstrate the clear outperformance of the proposed models in comparison with ϵ-ABRF.

Table 10. Comparison of ϵ-ABRF and the proposed models by using the measure R^2 when RFs and ERTs are the basis.

Data Set	RF		ERT	
	ϵ-ABRF	LARF	ϵ-ABRF	LARF
Diabetes	0.424	**0.434**	0.441	**0.471**
Friedman 1	0.470	**0.524**	0.513	**0.524**
Friedman 2	0.877	**0.933**	0.930	**0.933**
Friedman 3	0.686	**0.749**	0.739	**0.749**
Airfoil	0.843	**0.917**	0.837	**0.914**
Boston	0.823	**0.870**	0.838	**0.882**
Concrete	0.857	**0.896**	0.863	**0.896**
Wine	0.423	**0.481**	0.416	**0.486**
Yacht	0.989	**0.993**	0.988	**0.993**
Regression	0.450	**0.455**	0.447	**0.464**
Sparse	0.529	**0.641**	0.536	**0.663**

The best obtained results on each dataset are shown in bold.

It is obvious that the tuning parameters of the proposed modifications, for example, τ_0, are not optimal due to the validation procedure and due to the grid of values used in experiments. However, we can observe from the numerical results that the proposed modifications outperform RFs or ε-ABRF even with suboptimal values of the tuning parameters.

7. Conclusions

The new attention-based RF models proposed in the paper have supplemented the class attention models incorporated into machine learning models, differently from neural networks [7,8]. Moreover, the proposed models do not use gradient-based algorithms to learn attention parameters, and their training is based on solving the quadratic optimization problem with linear constraints. This peculiarity significantly simplifies the training process.

It is notable to point out that computing the attention weights in leaves of trees is a very simple task from the computational point of view. At the same time, this simple modification leads to the crucial improvement of the RF models. Numerical results with real data have demonstrated this improvement. This fact motivates us to continue developing attention-based modifications of machine learning models in different directions. First of all, the same approach can be applied to the gradient boosting machine [10]. The first successful attempt to use the attention mechanism in the gradient boosting machine with decision trees as base learners has been carried out in [7]. This attempt has confirmed that the boosting model can be improved by adding the attention component. The idea of the "leaf" attention and the optimization over parameters of kernels can be directly transferred to the gradient boosting machine. This is a direction for further research.

One of the important results presented in this paper is using a specific mixture of contamination models, which can be regarded as a variant of the well-known multi-head attention [16], where each "head" is defined by the kernel parameter. However, values of the parameter are selected in accordance with a predefined set. Therefore, the next direction is to consider randomized procedures to select the values of the parameter.

The proposed models consider only a single leaf of a tree for every example and implement the "leaf" attention in this leaf. However, they do not take into account neighbor leaves, which may also provide useful information for improving the models and should be studied.

It has been demonstrated in [8] that attention-based RFs allow us to interpret predictions by using the attention weights. The introduced two-level attention mechanisms may also improve the interpretability of RFs by taking into account additional factors.

Finally, we have developed the proposed modifications by using Huber's ϵ-contamination model and the mixture of the models. Another notable problem is to consider different available statistical models [46] and their mixtures. A proper choice of the mixture components may significantly improve the whole attention-based RF.

Author Contributions: Conceptualization, L.U. and A.K.; methodology, L.U. and V.M.; software, A.K.; validation, V.M. and A.K.; formal analysis, A.K. and L.U.; investigation, A.K. and L.U.; resources, A.K. and V.M.; data curation, A.K. and V.M.; writing—original draft preparation, L.U.; writing—review and editing, A.K. and V.M.; supervision, L.U.; project administration, V.M.; funding acquisition, V.M. All authors have read and agreed to the published version of the manuscript.

Funding: The research is partially funded by the Ministry of Science and Higher Education of the Russian Federation as part of the World-class Research Center program: Advanced Digital Technologies (contract No. 075-15-2022-311 dated 20 April 2022).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors would like to express their appreciation to the anonymous referees whose very valuable comments have improved the paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Chaudhari, S.; Mithal, V.; Polatkan, G.; Ramanath, R. An attentive survey of attention models. *arXiv* **2019,** arXiv:1904.02874.
2. Correia, A.; Colombini, E. Attention, please! A survey of neural attention models in deep learning. *arXiv* **2021**, arXiv:2103.16775. Available online: https://arxiv.org/abs/2103.16775 (accessed on 14 April 2023).
3. Correia, A.; Colombini, E. Attention, please! A survey of neural attention models in deep learning. *Artif. Intell. Rev.* **2022**, *55*, 6037–6124. [CrossRef]
4. Lin, T.; Wang, Y.; Liu, X.; Qiu, X. A Survey of Transformers. *arXiv* **2021**, arXiv:2106.04554.
5. Niu, Z.; Zhong, G.; Yu, H. A review on the attention mechanism of deep learning. *Neurocomputing* **2021**, *452*, 48–62. [CrossRef]
6. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]
7. Konstantinov, A.; Utkin, L.; Kirpichenko, S. AGBoost: Attention-based Modification of Gradient Boosting Machine. In Proceedings of the 31st Conference of Open Innovations Association (FRUCT), Helsinki, Finland, 27–29 April 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 96–101.
8. Utkin, L.; Konstantinov, A. Attention-based Random Forest and Contamination Model. *Neural Netw.* **2022**, *154*, 346–359. [CrossRef]
9. Friedman, J. Greedy function approximation: A gradient boosting machine. *Ann. Stat.* **2001**, *29*, 1189–1232. [CrossRef]
10. Friedman, J. Stochastic gradient boosting. *Comput. Stat. Data Anal.* **2002**, *38*, 367–378. [CrossRef]
11. Zhang, A.; Lipton, Z.; Li, M.; Smola, A. Dive into Deep Learning. *arXiv* **2021**, arXiv:2106.11342.
12. Nadaraya, E. On estimating regression. *Theory Probab. Appl.* **1964**, *9*, 141–142. [CrossRef]
13. Watson, G. Smooth regression analysis. *Sankhya Indian J. Stat. Ser. A* **1964**, *26*, 359–372.
14. Huber, P. *Robust Statistics*; Wiley: New York, NY, USA, 1981.
15. Utkin, L.; Konstantinov, A. Attention and Self-Attention in Random Forests. *arXiv* **2022**, arXiv:2207.04293.
16. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.; Kaiser, L.; Polosukhin, I. Attention is all you need. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 5998–6008.
17. Geurts, P.; Ernst, D.; Wehenkel, L. Extremely randomized trees. *Mach. Learn.* **2006**, *63*, 3–42. [CrossRef]
18. Liu, F.; Huang, X.; Chen, Y.; Suykens, J. Random Features for Kernel Approximation: A Survey on Algorithms, Theory, and Beyond. *arXiv* **2021**, arXiv:2004.11154v5.
19. Choromanski, K.; Likhosherstov, V.; Dohan, D.; Song, X.; Gane, A.; Sarlos, T.; Hawkins, P.; Davis, J.; Mohiuddin, A.; Kaiser, L.; et al. Rethinking Attention with Performers. In Proceedings of the 2021 International Conference on Learning Representations, Vienna, Austria, 3–7 May 2021; pp. 1–38.
20. Choromanski, K.; Chen, H.; Lin, H.; Ma, Y.; Sehanobish, A.; Jain, D.; Ryoo, M.; Varley, J.; Zeng, A.; Likhosherstov, V.; et al. Hybrid Random Features. *arXiv* **2021**, arXiv:2110.04367v2.

21. Ma, X.; Kong, X.; Wang, S.; Zhou, C.; May, J.; Ma, H.; Zettlemoyer, L. Luna: Linear Unified Nested Attention. *arXiv* **2021**, arXiv:2106.01540.
22. Schlag, I.; Irie, K.; Schmidhuber, J. Linear transformers are secretly fast weight programmers. In Proceedings of the International Conference on Machine Learning 2021. PMLR, Virtual, 18–24 July 2021; pp. 9355–9366.
23. Peng, H.; Pappas, N.; Yogatama, D.; Schwartz, R.; Smith, N.; Kong, L. Random Feature Attention. In Proceedings of the International Conference on Learning Representations (ICLR 2021), Vienna, Austria, 3–7 May 2021; pp. 1–19.
24. Brauwers, G.; Frasincar, F. A General Survey on Attention Mechanisms in Deep Learning. *arXiv* **2022**, arXiv:2203.14263.
25. Goncalves, T.; Rio-Torto, I.; Teixeira, L.; Cardoso, J. A survey on attention mechanisms for medical applications: Are we moving towards better algorithms? *arXiv* **2022**, arXiv:2204.12406.
26. Santana, A.; Colombini, E. Neural Attention Models in Deep Learning: Survey and Taxonomy. *arXiv* **2021**, arXiv:2112.05909.
27. Soydaner, D. Attention Mechanism in Neural Networks: Where it Comes and Where it Goes. *arXiv* **2022**, arXiv:2204.13154.
28. Xu, Y.; Wei, H.; Lin, M.; Deng, Y.; Sheng, K.; Zhang, M.; Tang, F.; Dong, W.; Huang, F.; Xu, C. Transformers in computational visual media: A survey. *Comput. Vis. Media* **2022**, *8*, 33–62. [CrossRef]
29. Kim, H.; Kim, H.; Moon, H.; Ahn, H. A Weight-Adjusted Voting Algorithm for Ensemble of Classifiers. *J. Korean Stat. Soc.* **2011**, *40*, 437–449. [CrossRef]
30. Ronao, C.; Cho, S.B. Random Forests with Weighted Voting for Anomalous Query Access Detection in Relational Databases. In Proceedings of the Artificial Intelligence and Soft Computing. ICAISC 2015, Zakopane, Poland, 14–18 June 2015; Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2015; Volume 9120, pp. 36–48.
31. Utkin, L.; Konstantinov, A.; Chukanov, V.; Meldo, A. A New Adaptive Weighted Deep Forest and its Modifications. *Int. J. Inf. Technol. Decis.* **2020**, *19*, 963–986. [CrossRef]
32. Winham, S.; Freimuth, R.; Biernacka, J. A Weighted Random Forests Approach to Improve Predictive Performance. *Stat. Anal. Data Min.* **2013**, *6*, 496–505. [CrossRef]
33. Xuan, S.; Liu, G.; Li, Z. Refined Weighted Random Forest and Its Application to Credit Card Fraud Detection. In Proceedings of the Computational Data and Social Networks, Shanghai, China, 18–20 December 2018; Springer International Publishing: Cham, Switzerland, 2018; pp. 343–355.
34. Zhang, X.; Wang, M. Weighted Random Forest Algorithm Based on Bayesian Algorithm. In *Journal of Physics: Conference Series*; IOP Publishing: Bristol, UK, 2021; Volume 1924, pp. 1–6.
35. Abellan, J.; Moral, S. Building Classification Trees Using th Building Classification Trees Using the Total Uncertainty Criterion. *Int. J. Intell. Syst.* **2003**, *18*, 1215–1225. [CrossRef]
36. Abellan, J.; Mantas, C.; Castellano, J. A Random Forest approach using imprecise probabilities. *Knowl.-Based Syst.* **2017**, *134*, 72–84. [CrossRef]
37. Abellan, J.; Mantas, C.; Castellano, J.; Moral-Garcia, S. Increasing diversity in random forest learning algorithm via imprecise probabilities. *Expert Syst. Appl.* **2018**, *97*, 228–243. [CrossRef]
38. Mantas, C.; Abellan, J. Analysis and extension of decision trees based on imprecise probabilities: Application on noisy data. *Expert Syst. Appl.* **2014**, *41*, 2514–2525. [CrossRef]
39. Moral-Garcia, S.; Mantas, C.; Castellano, J.; Benitez, M.; Abellan, J. Bagging of credal decision trees for imprecise classification. *Expert Syst. Appl.* **2020**, *141*, 1–9. [CrossRef]
40. Utkin, L.; Wiencierz, A. An imprecise boosting-like approach to regression. In Proceedings of the ISIPTA '13, Proceedings of the Eighth International Symposium on Imprecise Probability: Theories and Applications, Compiegne, France, 2–5 July 2013; Cozman, F., Denoeux, T., Destercke, S., Seidfenfeld, T., Eds.; SIPTA: Compiegne, France, 2013; pp. 345–354.
41. Utkin, L.; Kovalev, M.; Coolen, F. Imprecise weighted extensions of random forests for classification and regression. *Appl. Soft Comput.* **2020**, *92*, 1–14. [CrossRef]
42. Bahdanau, D.; Cho, K.; Bengio, Y. Neural machine translation by jointly learning to align and translate. *arXiv* **2014**, arXiv:1409.0473.
43. Luong, T.; Pham, H.; Manning, C. Effective approaches to attention-based neural machine translation. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Lisbon, Portugal, 17–21 September 2015; The Association for Computational Linguistics: NewYork, NY, USA, 2015; pp. 1412–1421.
44. Dua, D.; Graff, C. UCI Machine Learning Repository. 2017. Available online: http://archive.ics.uci.edu/ml (accessed on 20 April 2023).
45. Demsar, J. Statistical comparisons of classifiers over multiple data sets. *J. Mach. Learn. Res.* **2006**, *7*, 1–30.
46. Walley, P. *Statistical Reasoning with Imprecise Probabilities*; Chapman and Hall: London, UK, 1991.