Dynamic classification approach using scalable ensemble of autoencoders to classify data with drift

Anastasiya O Gurina[1,2], **Vladimir L Eliseev**[1,2] **and Sergey V Kolpinskiy**[1,2]

[1] National Research University "Moscow Power Engineering Institute", Krasnokazarmennaya st. 14, Moscow, 111250, Russia
[2] JSC InfoTeCS, Otradnaya st. 2B building 1, Moscow, 127273, Russia

asya.gurina001512@yandex.ru, vlad-eliseev@mail.ru and skolpinsky@bk.ru

Abstract. The problem of classification under concept drift conditions is investigated. The importance of anomaly detection is emphasized as a key feature of successful approach to operate with adversarial attacks and data poisoning. An approach to classification in the context of both drift and anomalies is introduced. It is based on ensemble of one-class classifiers, implemented by neural network autoencoders. Numeric parameters and supplementary logic are also supposed to distinguish between different classification cases. The quality of classifiers is estimated by original characteristics (EDCA), which examine both training set area and the area around it. The proposed approach is evaluated on synthetic data to highlight its properties in various conditions including normal, drift, new class and anomaly cases.

1. Introduction

Well-known classical machine learning methods such as support vector machine (SVM), decision trees, artificial neural networks demonstrate high classification quality. As a rule, classification is carried out under the assumption of stationary conditions, when the training data set fully describes the classification problem after the training sample has been obtained. However, this assumption is not always true. In non-stationary conditions, the training set may lose relevance if the objects of classification have changed. It is called concept drift.

Classification quality is usually measured by characteristics based on a confusion matrix. These usually include accuracy, precision, recall, and F1-score. However, these characteristics do not allow us to assess the quality of classification in the context of concept drift. Examples of adversarial attacks on neural network classifiers have also become widely known, in which examples are purposefully formed outside the scope of the training set, but are perceived by classifiers as correct. The problems of recognizing the drift of concepts and reducing the risk of adversarial attack are interrelated, since they require determining the behavior of the classifier outside the training set.

It seems important to develop an approach to constructing a classifier that allows, on the one hand, to track the concept drift, and on the other hand, to detect adversarial attacks.

Section 2 of the article provides a review of sources on issues related to classification in a drift conditions. Section 3 proposes a method for detecting drift, new classes and anomalies using autoencoders, as well as a method for assessing the vulnerability of trained models. Section 4 demonstrates the results obtained when testing the proposed approach on synthetic data. Section 5

contains discussion of the contribution. Section 6 concludes the article and indicates possible directions for future research.

2. Background

Non-stationary conditions are characterized by the problem of data drift. Data drift means that the underlying data distribution changes over time [1]. As a result, the input data has changed in general and the trained model is not relevant for this new data. There are different types of drift. It can be gradual, sudden, and recurring (seasonal). Gradual drift means that the probability of the old data distribution will decrease, and the probability of a new distribution will increase during a period of time until the new distribution substitutes the old one. Any drift causes static model decay.

The drift problem may occur in various real-world applications, for example:

- in face recognition systems,
- in text classification systems, spam,
- when identifying a user by behaviour,
- computer systems or networks when classifying network intrusions,
- in industry when classifying the state of a plant and many others.

In recent years machine learning researchers are faced with the problem of data drift when mining and classifying non-stationary data streams. This has led to the development of tailored approaches that extend the capabilities of traditional machine learning methods. Most often, the current accuracy of the model is monitored to detect drift of any type. If it decreases, then it means that the model becomes invalid and needs to be modified [2], [3], [4]. It is also possible to monitor changes in the statistical properties of the data itself [5], [6]. An autoencoder is a promising tool for detecting different types of drift [7]. Typically, after a drift is detected, the classifier is retrained on the current data on the assumption that it better describes the actual distribution of the data.

Sometimes the drift detection mechanism is included in the data stream classification algorithm itself. There are three main groups of such approach: incremental learning based approaches [8], [9], window-based approaches [10], and ensemble-based approaches [11], [12], [13].

The most popular evolving technique for handling concept drift in data streams is to use an ensemble classifier (a combination of classifiers), such as in [12]. The outputs of multiple classifiers are combined to determine the final classification, which is often called fusion rules.

The problem of poisoning the training data with anomalies can be solved by a classifier with adaptation to new data [11] [14] [15].

Some of the approaches mentioned above solve the problem of detecting the concept drift, and some - detecting anomalies due to adversarial attacks [16], but none of the approaches allows detecting both situations within the framework of a unified approach. In addition, some of the considered approaches are not protected from training on data contaminated with (poisoning) anomalies, which allows an attacker to form a classifier for his own purposes. To control the concept drift and anomalous data, an approach based on an ensemble of autoencoders seems promising.

3. Dynamic classification approach using scalable ensemble of autoencoders

3.1. Detection of drift, new classes and anomalies using an autoencoder

The proposed method is based on an algorithm for detecting novelty in data using a neural network autoencoder [17], [7]. The training of the autoencoder continues until it reconstructs samples of the training set with acceptable accuracy. The output of the autoencoder is called reconstruction.

The trained autoencoder can be used to measure the closeness of the input samples to the training data. The autoencoder is often used as a one-class classifier. The degree of closeness of the input sample to the training data is determined by the value of *immediate reconstruction error* (IRE). The IRE value for the input sample $X(x_1, x_2, \dots, x, \dots x_m)$ of dimension m is calculated by the formula: $IRE_X = \sqrt{\sum_{i=1}^{m}(x_i - y_i)^2}$, where $Y(y_1, y_2, \dots, y, \dots y_m)$ is the reconstruction of the input sample. The closer IRE_X is to zero, the more accurately the autoencoder has reconstructed the input sample and the more reliable

the hypothesis that the input sample belongs to the training set. To detect novelty in the data, the threshold value IRE_{th} is determined. The threshold value in this paper is chosen as the maximum value among the reconstruction errors calculated for the samples of the training set.

The threshold value of the reconstruction error can be interpreted as a class boundary in the feature space. Thus, if the IRE_X for a sample X exceeds the recognition threshold IRE_{th}, it means that the sample is outside the class boundary.

In the real world, it is important to distinguish between two cases: the outlier sample is near the class boundary or it is far from it. Let's consider the IRE as a metric of the closeness of a sample X to the class boundary, defined by $IRE_X \leq IRE_{th}$. An appearance of new samples outside, but nearby the class boundary over time may indicate a gradual drift of the class. In this case, it makes sense to retrain the autoencoder with the inclusion of newly received samples in the training set. The appearance of a new sample X at a significant distance from the boundary ($IRE_X \gg IRE_{th}$) may indicate the appearance of a new class or an anomaly. Both cases do not need retraining of this classifier.

To distinguish concept drift from other cases, a coefficient of proportional expansion of the boundary $k_{drift} > 1$ is introduced.

The application of the introduced logic is formally described in table 1 and illustrated in figure 1. However, the shape of outer boundary of drift area may be more complex than a simple expansion of the class boundary in feature space.

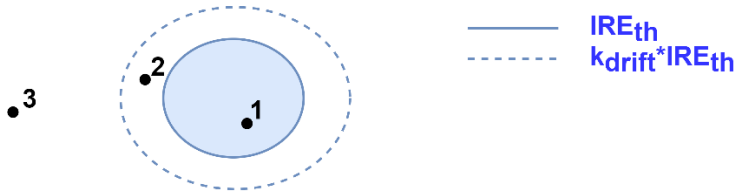


Figure 1. Cases of classification using a single autoencoder.

Table 1. Cases of classification using a single autoencoder.

Case	Conditions	Classification result	Reaction method
1	$IRE_{X1} \leq IRE_{th}$	Sample of a known class	Nothing extra
2	$IRE_{th} < IRE_{X2} \leq k_{drift} IRE_{th}$	Sample of a known class with a drift	Retraining on new data
3	$IRE_{X3} > k_{drift} IRE_{th}$	Sample of a new class or an anomaly	Extra processing

In the case of a one-class classification, it is possible to determine whether a sample is an anomaly if its IRE is above some boundary, which can be defined as $k_{anom} IRE_{th}$. Thus, if $IRE_{X3} > k_{anom} \cdot IRE_{th}$, then this sample can be classified as an anomaly.

One can consider $k_{drift} = k_{anom}$ to detect anomaly as a boundary of acceptable drift. However sometimes one needs to distinguish a new class case from the significant anomaly. The gap between anomaly and drift detection can be applied as a new class condition if $k_{drift} < k_{anom}$: $k_{drift} IRE_{th} < IRE_{X3} \leq k_{anom} IRE_{th}$.

The case of a new class detection should be processed as a new classifier introduction, which needs data gathering for its training. The case of an anomaly needs reporting only and does not affect classifiers.

3.2. Data classification under non-stationary conditions using an ensemble of autoencoders.

The proposed method of using an autoencoder to operate with one class can be extended to the case of multiple classes. To solve the problem of multiclass classification with the drift capability, new data and anomaly detection, an ensemble of autoencoders can be used. Let's assume the training set consists of samples of n known classes $C_1, C_2, \dots, C_n$. It is proposed to build an ensemble of n autoencoders $AE_1, AE_2, \dots, AE_n$. They should be trained to recognize samples of one of the known classes

$C_1, C_2, \dots, C_n$, as shown in Section 3.1. Also one more autoencoder AE_0 is introduced. It should be trained on all samples of training sets of all classes: $C_0 = C_1 \cup C_2 \cup \dots \cup C_n$. For trained autoencoders, the threshold values IRE_{th_j} $(j = 0,1, \dots, n)$ can be determined just the same way as for one class in Section 3.1.

The threshold values of the reconstruction error determined in this way for each autoencoder make it possible to outline n areas of known classes $C_1, C_2, \dots, C_n$ in the feature space. As well as the area C_0 overlying all known classes (figure 2). If the autoencoder AE_0 is trained qualitatively, then we can expect that the C_0 area will cover the known classes quite densely.

Exactly the same k_{drift} coefficient can be defined to detect concept drift for every class by using corresponding autoencoder IRE_{th_j}, where $j = 1,2, \dots, n$. Figure 2 gives several different cases of sample location in the feature space. For example, points 2a and 2b represent concept drift for C_1. In case of drift detection one needs to retrain affected autoencoders.

Several autoencoders may accept the same sample as their own belonging. A class with AE_j which gives minimum of the ratio IRE_{X_j}/IRE_{th_j} is defined as a matching one. The same rule can be applied for drift detection by several autoencoders: the class for which the concept drift is solved to be detected marks the minimum excess over IRE_{th_j}among all.

In the cases discussed earlier, it was assumed that the known classes could be expanded. It is also worth considering the case when a gradual, recurring or sharp class drift is detected. Then, over time, some of the data on which the autoencoder was initially trained will become irrelevant for this class. Such data should be forgotten. A full retraining of the autoencoder is possible when the necessary amount of relevant training data is collected.

Now let us consider the case when the previous steps of the classification algorithm define that a new sample does not belong to known classes. Then the threshold criterion $k_{anom}IRE_{th0}$ is used to distinguish the cases of the appearance of a new class or anomaly. If the IRE for the new sample does not exceed $k_{anom}IRE_{th0}$, then this sample is classified as a sample of new class (case '3' in figure 2). In this case, a new autoencoder AE_{n+1} should be created and trained. As well as retrain AE_0 taking into account the samples of the new class. If the IRE for a new sample exceeds $k_{anom}IRE_{th0}$, then this sample is classified as an anomaly (case '4' in figure 2). If an anomaly is detected, it is recommended to generate an alerting signal. For many subject areas, it is critically important. Various possible cases are formally described using the conditions in table 2 and illustrated in figure 2.

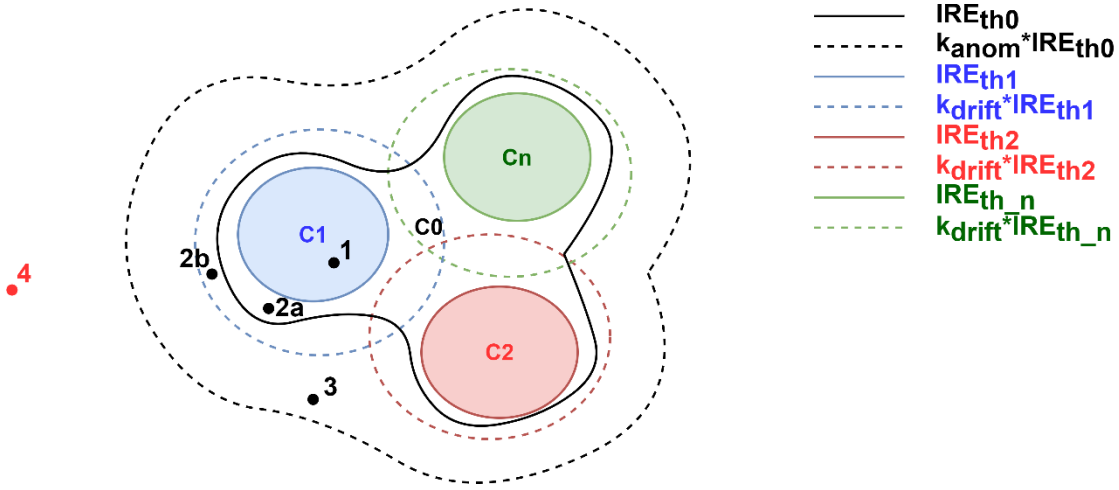


Figure 2. Cases of classification using an ensemble of autoencoders.

Table 2. Cases of classification using an ensemble of autoencoders.

Case	Conditions	Classification result	Reaction method
1	$IRE_{X1} \le IRE_{th(1\|2\|\dots n)}$	Sample of a known class with the least ratio IRE_{X1}/IRE_{th_j}	Nothing extra
2a	$IRE_{th(1\|2\|\dots n)} < IRE_{X2a} \le k_{drift} IRE_{th(1\|2\|\dots n)}$ $and \;\; IRE_{X2a} \le IRE_{th0}$	Sample of a known class with drift	Retraining $AE_{(1\|2\|\dots n)}$ on new data
2b	$IRE_{th(1\|2\|\dots n)} < IRE_{X2b} \le k_{drift} IRE_{th(1\|2\|\dots n)}$ $and \;\; IRE_{th0} < IRE_{X2b} \le k_{anom} IRE_{th0}$	Sample of a known class with drift	Retraining $AE_{(1\|2\|\dots n)}$ and AE_0 on new data
3	$IRE_{X3} > k_{drift} IRE_{th(1\|2\|\dots n)}$ $and \;\; IRE_{th0} < IRE_{X3} \le k_{anom} IRE_{th0}$	Sample of new class	Creating and training AE_{n+1}, retraining AE_0 on new data
4	$IRE_{X4} > k_{anom} IRE_{th0}$	Anomaly	Extra processing

In general, the contribution of this paper is a simple and effective method for classification and anomaly detection data with drift. Further, the classifier based on the proposed approach is called scalable ensemble of autoencoders (SEAEs).

The approach also includes an evaluation of the accuracy of the approximation of the training data areas by the model using new characteristics. The method of evaluating the vulnerability of a trained model to adversarial attacks is described in the next section.

3.3. Assessing adversarial vulnerability using new characteristics.

To enhance the proposed approach against adversarial attacks the approximation accuracy of the training data area should be estimated. The trained classifier approximates the training data area in the feature space. If the approximated area completely matches the training data area, then the model is not vulnerable to adversarial attacks (figure 3b). Otherwise (figure 3a), there is an adversarial example that can cheat the classifier.

An original approach to estimate approximation accuracy of the training data area was proposed in research paper [18], where four new characteristics were introduced: Excess, Deficit, Coating, Approx (EDCA). They are calculated from two discrete estimates in feature space. The first estimate is a discrete volume that the training data occupies in feature space $|X_T^*|$. The second is an estimate of the volume in the feature space where data are recognized by the classifier. Since this volume differs from training set volume, it can be called a deformed set volume $|X_D^*|$. To determine the data under each class, the feature space is scanned with some regular step. The scanned discrete points are classified using the trained model to evaluate its belonging to the area of classification. The area of points with an exact classification result forms a discrete volume under the resulting class. For a multi-class classifier one needs to calculate the mentioned discrete volume estimates for each class.

To estimate the data volume, the feature space is divided into atomic cells. The cell of X_T^* is not empty if it contains at least one point from training set. The cell of X_D^* is not empty if at least one point matches the class from the classifier point of view.

New characteristics are calculated on the basis of discrete estimates of training $|X_T^*|$ and deformed $|X_D^*|$ set volumes using the following formulas:

$$Excess = \frac{|X_D^* \backslash X_T^*|}{|X_T^*|} \qquad Deficit = \frac{|X_T^* \backslash X_D^*|}{|X_T^*|}$$

$$Coating = \frac{|X_T^* \cap X_D^*|}{|X_T^*|} \qquad Approx = \frac{|X_T^*|}{|X_D^*|}$$

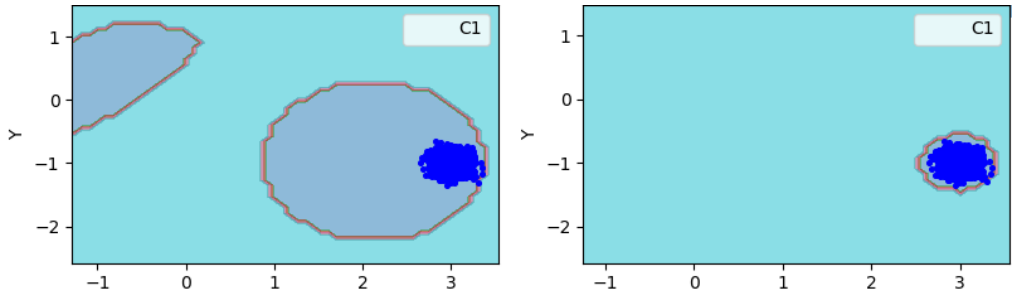
An ideal one-class classifier model should have the following characteristics:

$$Excess = 0, \quad Deficit = 0, \quad Coating = 1, \quad Approx = 1$$

The proposed criterion can be easily generalized to any number of classes. In such case every characteristic should be defined for its own class separately.

In the traditional approach, the quality of the classifier is evaluated by the classification results of the test set. Moreover, the quality assessment of the classifier is based only on the aggregate characteristics for the classifier as a whole (accuracy, precision, recall, F1-score). Let's evaluate the quality of the classifier by EDCA criterion. To apply the method it is enough to have only a training set and to know the reasonable limits of all dimension of features.

Let us demonstrate new characteristics effectiveness by an example. Consider two one-class classifiers AE1 and AE2. The first autoencoder AE1 has the architecture [2; 1; 2], and the second autoencoder AE2 has the architecture [2; 3; 5; 2; 1; 2; 5; 3; 2]. Autoencoders AE1 and AE2 are trained on the same synthetic dataset for 1000 epochs. Figure 3 shows the areas of points in the feature space that trained autoencoders assigned to the target class.



(a) (b)

Figure. 3. Class boundaries constructed by autoencoders AE1 (a) and AE2 (b)

Figure 3a shows that the first autoencoder AE1 built several target areas, while the training data is in only one of them. Such an autoencoder is vulnerable to adversarial attacks. The AE2 autoencoder architecture is more complex, so it approximates the training data area more accurately. To demonstrate the consistency of the criterion, table 3 shows the values of the proposed characteristics for the first and second autoencoders.

Table 3. Results of assessing the vulnerability of trained models

Classifier	Excess	Deficit	Coating	Approx
AE1	11.25	0	1	**0.08**
AE2	0.5	0	1	**0.67**
Ideal AE	0	0	1	**1**

The values for the autoencoder AE2 are closer to ideal. That means it is less vulnerable to adversarial attacks, as shown in figure 3b. Thus, the new characteristics allow to assess the actual resistance of the trained model to an adversarial attack. The proposed characteristics are especially important in the case of high data dimensions. The high dimensionality of the feature space makes it impossible to accurately visualize the area that the classifier also assigns to the target class, as in figure 3. Thus, testing of the trained models using EDCA characteristics makes it possible to build more reliable classifiers.

4. Experimental results

In this section, we present results of SEAEs testing conducted on simple synthetic datasets with drift, anomalies and new class data in test sets.

The training data contains only two data classes. The dimension of the training data is D=400. The two test sets T1 and T2 are constructed in such a way, that the concept drift occurs after 400 data samples processing. The test dataset T1 additionally includes samples of new class. And test dataset T2

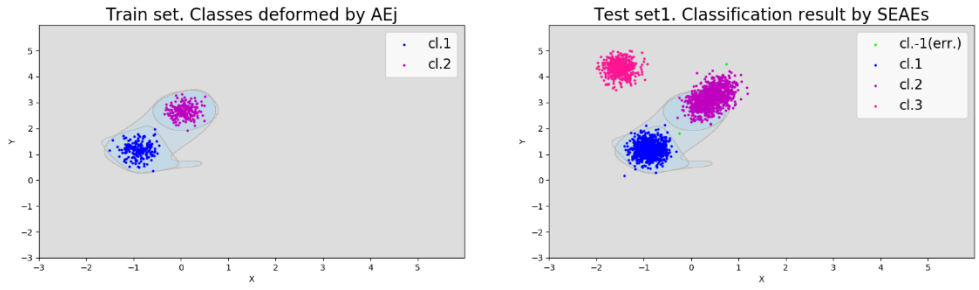
additionally includes anomaly samples. In the experiments, we used a 7-layered autoencoder for each class. The dimensions of the input and output layers match the dimensionality of the data. The dimension of the middle layer is set to 4. The AE_0 autoencoder has the same architecture. The ADAM learning algorithm is used for training. The learning rate is set to η=0.01. To implement the SEAEs classifier, parameters k_{drift} and k_{anom} are set to 1500 and 10000, respectively.

In the first experiment, the ensemble of autoencoders is trained only for the first 400 data elements. To assess the vulnerability of the trained SEAEs, the values of new characteristics (EDCA) were calculated for each autoencoder. The values of the new characteristics are calculated using formulas (1-4) given in Section 3.3. The results of the vulnerability assessment of the trained models are shown in Table 4.

Table 4. Results of vulnerability assessment of trained models

SEAEs	Excess	Deficit	Coating	Approx
AE1	0.47	0	1	**0.68**
AE2	0.77	0	1	**0.57**
AE0	0.94	0	1	**0.52**

The obtained values make sure that the feature space does not contain large areas that may include adversarial examples. After a successful check, the SEAEs switches to the monitoring mode. The classification was carried out in accordance with the rules shown in table 2. Examples of the training set, as well as the class boundaries constructed by the autoencoders in the feature space, are shown in figure 4a. The result of the classification of test set T1 using a SEAEs is shown in figure 4b.



(a) (b)

Figure 4. Result of test set T1 classification using SEAEs

The SEAEs classified with high accuracy the drifting data of the class 2 and the data of the new class (cl.3). This is also confirmed by the traditional characteristics of classification quality (table 5).

Table 5. The SEAEs assessment results by traditional characteristics

Classifier	Precision	Recall	F-score
SEAEs	1	0.99	0.99

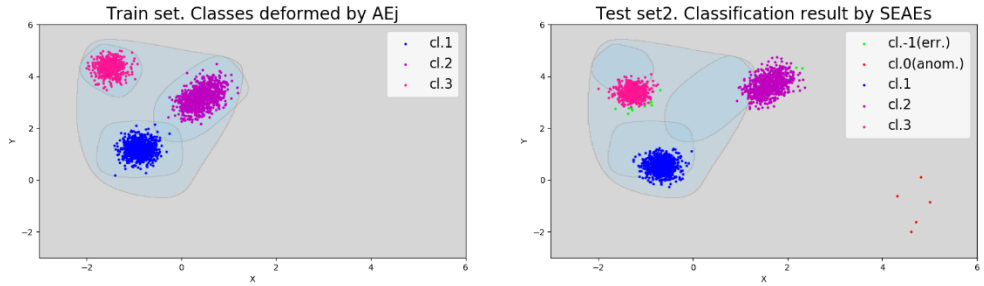
The classifier was retrained on data without anomalies when enough examples were collected. The ensemble classifier pool was expanded to accommodate the new class. For class 3, a new autoencoder of the same architecture was created. The training sample for retraining consisted of up-to-date data. After retraining, the stability of the classifier was also assessed using new characteristics (EDCA). The results of the model evaluation after retraining are presented in table 6.

Table 6. Results of vulnerability assessment of trained models (after retraining)

Re-trained SEAEs	Excess	Deficit	Coating	Approx
AE1	0.76	0	1	**0.57**
AE2	1.06	0	1	**0.49**
AE3	0.47	0	1	**0.68**
AE0	1.15	0	1	**0.46**

The characteristic values show that the class boundaries built by autoencoders have become wider. This is because the training data for retraining is more varied. This is confirmed by figure 5a.

Figure 5b shows the classification results for the T2 test set, which includes data drift and anomalies.



(a) (b)

Figure 5. Result of test set T2 classification using SEAEs

A timely retrained classifier successfully classified drift data and anomalies. This is confirmed by the values of traditional classification quality characteristics, which are shown in table 7.

Table 7. The SEAEs assessment results by traditional characteristics

Classifier	Precision	Recall	F-score
Re-trained SEAEs	1	0.99	0.99

The obtained experimental results confirm the effectiveness of the proposed dynamic classification approach using SEAEs.

5. Discussion

The dynamic classification approach provides next advantages:

- universality: normal, concept drift, new class and anomaly cases are handled;
- economy of resources: it is not necessary to retrain all ensemble classifiers, if a drift of one class is detected;
- low risk of data poisoning: filtering gathered data on anomalies.

However, a lot of resources are needed to train one-class classifiers for a large number of classes.

6. Conclusions

In this article, we focus on the dynamic classification of data streams under non-stationary conditions. We propose a new approach based on a scalable ensemble of autoencoders. The approach allows for the emergence of new classes, drift, anomalies and adversarial attacks in classification. The results of experiments on synthetic data show the effectiveness of the proposed approach. In future work, we plan to test the proposed algorithm on real high-dimensional data streams.

Acknowledgments

The reported study was funded by RFBR according to the research project № 20-37-90073.

References

[1] Webb G, Hyde R, Cao H. et al. 2016 Characterizing concept drift *Data Min. Knowl. Disc.* **30** 964–994

[2] Gama J, Medas P, Castillo G, Rodrigues P 2004 Learning with drift detection *Intelligent Data Analysis* **8** 286-295

[3] Baena-Garcia M et al. 2006 Early drift detection method **6** 77–86

[4] Mashail S A, Manal A 2020 CDDM: Concept Drift Detection Model for Data Stream *iJIM* **14** 90-106

[5] Dong F, Zhang G, Lu J, Li K 2018 Fuzzy competence model drift detection for data-driven decision support systems *Knowl. Based Syst*. **143** 284–294

[6] Boracchi G, Carrera D, Cervellera C, Maccio D 2018 Quanttree: Histograms for change detection in multivariate data streams *Conf. on Machine Learning, Stockholm, Sweden* 10–15 639–648.

[7] Jaworski M, Rutkowski L, Angelov P 2020 Concept drift detection using autoencoders in data streams processing *Artificial Intelligence and Soft Computing* (Cham: Springer Int. Publishing) **12415** 124-133

[8] Gregory D 2013 Incremental learning of concept drift from imbalanced data *Know. Data Eng.* **25** 2283-2301

[9] Krawczyk B, Michał W 2015 One-class classifiers with incremental learning and forgetting for data streams with concept drift *Soft Computing* 3387-3400

[10] Bifet A, Gavald´a R 2007 Learning from time-changing data with adaptive windowing *7th SIAM International Conf. on Data Mining* **7** 443–448

[11] Sun Y, Shao H, Wang S 2019 Efficient ensemble classification for multi-label data streams with concept drift *Information* **10**(5) 158

[12] Sarnovsky M, Kolarik M 2021 Classification of the drifting data streams using heterogeneous diversified dynamic class-weighted ensemble *PeerJ Comput. Sci.* **7**

[13] Ludwig S A 2019 Applying a neural network ensemble to intrusion detection *J. Artif.Intelli. Soft Comput. Res.* **9**(3) 177–188

[14] Tennant M, Stahl F, Rana O, Gomes J B 2017 Scalable real-time classification of data streams with concept drift *Future Generation Comp. Sys.* **75** 187-199

[15] Togbe M U, Chabchoub Y, Boly A, Barry M, Chiky R, Bahri M 2021 Anomalies detection using isolation in concept-drifting data streams *Computers* **10**(1) 13

[16] Wiggers K 2021 Adversarial attacks in machine learning: What they are and how to stop them?

[17] Елисеев В Л, Гурина А О 2019 Нейросетевой метод выявления новизны в модели нестационарной динамической системы *XXXIII Международная научно-техническая конф. Проблемы автом. и упр. в тех. сист.* **2** 237-241

[18] Гурина А О, Елисеев В Л 2021 Эмпирический критерий качества одноклассового классификатора *XXVII международная научно-техническая конф. Инф. Сист. и технол.*