A Concept of Continuous User Authentication Based on Behavioral Biometrics

Aleksandr Eremin, Konstantin Kogos, Alina Filina
NRNU MEPhI
Moscow, Russia
ave_38@mail.ru, kgkogos@kaf42.ru, anfilina@hotmail.com

***Abstract*—This paper focuses on continuous user authentication based on its interaction with the device. Behavioral authentication provides the ability to partially abandon passwords. Furthermore, the use of human behavior, for example, how he holds the device in hand, interacts with a screen, as a means of authentication is sufficiently protected from compromise, since an attacker cannot make an exact copy your gait or motion. Use of auxiliary factors such as the proximity of the trusted peripheral device, Wi-Fi network, location, helps simplify authentication.**

I. Introduction

Mobile devices store a lot of personal information about its users. Some sensitive information compromise such as correspondence in social networks, texting, calls, photos, etc. can do harm to user's reputation. It is clear it is necessary to protect this data from unauthorized. Users question is which way to protect confidential information is the best.

It is generally agreed that there are three basic approaches for user authentication. The first is based on the fact that the user knows, the second on what he has, and still others on what is inherent in him. The knowledge factors mean that the person knows a password, PIN, unlock pattern, etc. The ownership factors imply that the user has something that could confirm his identity, for example peripheral devices. Finally, the inherence factors. Such features as the behavior when interacting with the device, fingerprint and face are unique to a user and can be used for authentication. This approach can be called a biometric.

Password remains the most common way to authenticate a user. Typically used password made up of four or six digits. Such protection can be easily bypassed if the intruder spied on the entered combination. The same applies to the unlock pattern. When a person moves a finger across the screen device he leaves grease marks on it. There is also a side channel, which allows to determine the user-entered information using the accelerometer built into the mobile device [1]. Authentication using the peripheral devices such as a fitness tracker or a smart watch is also not reliable. To pretend to be a legal user, the intruder is only to seize the peripheral device, or simply use the smartphone when its owner is not far from it. It can be seen, that it is not difficult to get access to confidential data on a smartphone or tablet, if the device is protected with something that the user knows or possesses. Biometric approach to information protection consists in using person's unique characteristics which are almost impossible to forge. Authentication methods based on biometrics are characterized by a high degree of security, rapidity of the process and ease of use.

Various aspects can be used for biometrics authentication. The device can identify the user by physical or behavioral characteristics. The biometric authentication based on the physical properties of entity is the most common.

More and more devices use fingerprint scanner for authentication. This method has already proved its safety and convenience, but the sensor does not always work, for example, if the user's hands are wet, the smartphone offers to read the fingerprints again, or enter a password. In this case, this method of determining the individual takes too much time, causing a lot of troubles to a smartphone user.

There are methods of using a person's face to determine his identity. But their use can be limited by the low-light room, a bad turn of the head and other factors. Authentication by voice is another well-known method of biometric authentication. Unfortunately, these two methods are very easy to bypass by putting a photo of the owner in front of the camera or turning on pre-recorded voice. Therefore, it is rather difficult to use them in practice properly.

Continuous authentication allows you to grant rights to the user, without requiring from him any unusual activities. The use of continuous authentication allows partially abandon passwords and other authentication methods, which consume a certain time. Deceive behavior-based authentication is almost impossible. There is an attack on other methods using biometric data, but it is not possible to fake user behavior or some features of his interaction with the smartphone.

Behavioral biometrics can provide keen and tough competition with a fingerprint scanner and a password. Most important advantage is that the user does not perform any additional actions, he simply uses the device normally. This article will be devoted to the behavioral authentication.

The remainder of this paper is organized as follows. Section II contains an overview used background and related work. Section III introduces a proposed architecture of an application that provides continuous user authentication. An overview of machine learning algorithms that can be used to decide whether the user is an authorized user is described in Section IV. Conclusions and further research are presented in Section V.

ISSN 2305-7254

II. Background and related works

The article is based on several behavioral user characteristics, which are presented below.

Most of the phones in the world are equipped with a touch screen. This sensor provides us with a big amount of information about how we touch the screen, or touch dynamics. In [2] is shown that while the user is making a swipe, the system observes pressure (how hard the finger presses), size (area of the finger touching the screen), coordinates, and time of contact. Besides these parameters acceleration is measured with the help of accelerometer. All these parameters are changing while user is interacting with the touch screen, drawing a pattern, for example. The data gathered is processed with a machine learning algorithm that helps to make a conclusion whether the user is an owner of the phone. It is possible due to the uniqueness of swipes that different users make [3]. Such approach is used in implicit or continuous authentication and allows to invisibly provide security for a user.

Keystroke dynamics analysis goes along with the analysis of the touch in modern mobile devices, as they have a virtual keyboard on the screen, so the same sensors are involved. The parameters of particular interest are:

- time interval between releasing a key and pressing of the next key;
- interval between two keystrokes;
- time passed between pressing and releasing a single key;
- number of mistakes (how many times Backspace is pressed);
- distance between two keystrokes in pixels;
- speed, computed from time between keystrokes and distance between this keys.

According to [4] two approaches to receiving data are static and dynamic. In static typing, user is asked to type a predefined text to compare motion information with previous results, while in dynamic typing, the subject is free to type any text. Authors of [4] note that commonly used statistical classifiers found in the literature include Bayes (and naive Bayes), Mahalanobis distance, Hamming distance, Euclidean distance, etc. More recently, neural networks have been used as a pattern classification method. Common neural network approaches include Feed Forward Multilayered Perceptron Networks (with and without back propagation), Radial Base Function Networks and Generalized Regression Networks.

Gait dynamics-based authentication method also relates to continuous authentication methods. It helps to recognize users by how they walk. There are three main ways to measure necessary parameters:

- machine vision based;
- floor sensor;
- wearable sensor.

The first two approaches are not applicable in smartphones, but the third one is a useful method as all modern smartphones are equipped with accelerometer and gyroscope.

Data obtaining can be cyclic or non-cyclic. The first method consists of two steps. First of all, “cycles” in gait are identified. Features of these cycles are processed to extract characteristic templates for classification. A non-cyclic approach captures locations of sensors in specified time intervals during the walking [5]. The main difficulty in data analysis is that human can walk with different speed, he can run, jog, climb stairs, etc. In that case, special measurements for every type of gait are held. That helps to improve the precision of the results; it is shown in [6], where the best accuracy was brought by SVM.

Jakobsson et al. in [7] show that modern mobile devices provide us with comprehensive amount of data about user’s habits and behavior patterns. Authors used several parameters to verify the user:

- location and co-location data from GPS;
- WiFi/Bluetooth connections and USB connection to a known PC;
- application usage, such as browsing patterns and software installations;
- log of calls, SMS, etc;
- contextual data, such as the contents of calendar entries, the current time of day, day of week, etc.

All this data helps the authors to perform an implicit authentication, using scoring algorithm: having a model model of recent behavior, the scoring algorithm outputs a score representing the likelihood that the device is used by the rightful owner.

Zhu et al. [8] propose a mobile framework model Sensec based on accelerometer, orientation, gyroscope, and magnetometer to construct a user gesture profile and use it to continuously authenticate a user. The model continuously computes the score authenticating the user. A valid user was identified with 75% accuracy and an adversary with an accuracy of 71.3% from a set of 20 users. However, this research requires a user to follow a fixed sequence of actions and collects data for the entire user interaction session. Li et al., [*9*] studied the ability of using three different sensors: accelerometer, orientation, and compass in addition to the touch gestures for continuous user authentication. Their method obtains finger movements using classical touch-based features and interprets the collected data as different gestures. An SVM classifier is than trained with gestures to perform user authentication. Accuracy of 95.78% was gained using a database of 75 users.

III. Architecture

Standard authentication on modern devices is performed before the user is going to use the device for his own purposes. At the same time, absolutely unimportant what he wanted: make a call, send a message, use a calculator or listen to

music. Any authentication method takes some time. Often it is this factor which makes the users refuse the phone protection, as a result, the risk of damage to the owner's data significantly increases. A compromise in these situations is a method to authenticate a user only when he wants to use certain possibilities of smartphone or tablet.

The idea is that the authentication is not carried out before using the functions of the phone, and the user, regardless of whether he is legal, has access to certain device features. And the rest is possible only after authentication.

Continuous authentication allows to authenticate the user before it tries to gain access to critical applications. At the same time, the person cannot even imagine what he has already introduced his authentication data. The data for authentication are biometric and are based on user behavior, i.e. his interaction with the smartphone.

Let authentication fails, and the user has been identified as an intruder. Then, when he tries to open the protected application the password prompt will be displayed. If entered password is wrong the timeout will be set for the following input for access not only to the chosen, but to each protected application. In case, if the user was authenticated, then the application will be opened successfully.

At the first start the application, that is responsible for the authentication, must be configured. The user sets the Master Password that will allow to manage the applications, and authentication password, that is requested if the user has not been identified as legitimate. The application has a section called My Account. In this section, Wi-Fi networks are set. No authentication is required when a device is connected to them, let us call such a network the Trusted one. The user can also add a few peripherals that connect via Wi-Fi, Bluetooth or USB-interface. When you connect your smartphone to a PC, this computer can also be added to the Trusted devices. Then, if the smartphone is associated with Trusted device identity verification is also not carried out. In section My location a few areas can be added where you often see yourself and trust to all who are in them. For example, House, Cottage, Parents' House.

The purpose of the program is to protect the specific functions and programs of the device. By default, security is enabled for applications Contacts, Calls, Messages, Browser, File Explorer (My Documents), Settings, Photo, App Store. The rest of the applications, the user adds to the list of Protected Applications as desired. There the default settings can also be changed. Turning off the protections Settings can lead to involuntary service termination of the Application, so this change cannot be performed.

In the Application, the function "Do not authenticate during ..." is available. This feature allows the user to give the device to another person to use for a predetermined period of time. Authentication will not be performed at this time, and data obtained from the sensors are not added to the sample corresponding to the user.

In order to properly authenticate a person, it is necessary to collect enough information about him. In this case, these are behavioral properties, which are the biometric data. For some time, information is collected about the Wi-Fi networks and Bluetooth devices, to which the user connects, and where he is. In addition, the data are collected from the touch screen, accelerometer and gyroscope. The obtained data are processed in such a way that had no spurious emissions, inaccurate measurements. So, pre-processing is carried out. Based on the data taken, speed, acceleration, the contact area of the finger to the screen, the interaction region the finger and sensor of the smartphone, phone position in space are calculating.

The collected data of the trainee period forms a set of objects for the training sample of machine learning algorithm. Later it will determine the affiliation of new data to the user. After completion of the training period of the owner recognition the exact use of authentication Application starts.

User authentication only happens when interacting with the device. Fig. 1. shows the logic of the application work. The first measurement is carried out immediately when the screen is turned on. First, it is checked whether a device is connected to the smartphone and, if connected, whether it is the Trusted one. If it is determined that the trusted device is in the network coverage area, i.e., the user is near, it is considered successfully authorized. If the Trusted Devices are not available, or are not connected to the smartphone, there is a check on the other parameters.

An inspections of Wi-Fi connections is carried out. Similarly, like the Trusted Device, it is checked whether the smartphone is connected to a network, if so, whether it is a Trustee. If the device is in a trusted network of Wi-Fi, then the user is considered to be successfully authenticated. In the case where the device is not connected to any network, or connected to the not trusted network, there is a further check on the user's identity.

By using geolocation services the location of the device is determined. After the program compares the value of a list of Trusted Locations with the current location the user is authenticated or the next phase of reading information begins.

The data is read from the accelerometer, gyroscope, and a touch screen. As well as in the learning phase, the measured data is converted into behavioral characteristics that form the inputs to a machine learning algorithm. The algorithm receives the input of new objects and determines whether the received data user corresponds to a legal user. Details about methods of machine learning will be discussed in Section IV.

If the algorithm has determined that the new data is not characteristic to the legal user, the person who uses the device is supposed an attacker. If he tries to open a protected application, a popup will appear, prompting him to type the answer on the secret question that was set in My Account when the initial applications setup was done. If the answer is wrong, the password is asked. If the correct password is introduced, the user is authorized. Otherwise, the data collected from the sensors is included in the sample as an anomaly. The access to designated applications is blocked for a certain period *t*.

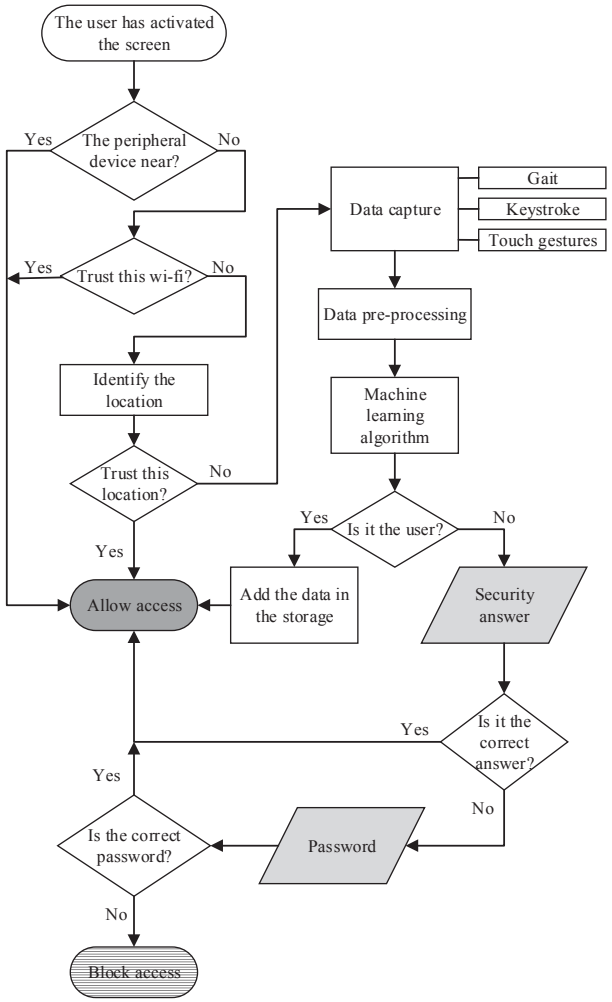

Fig. 1. Proposed work logic of the application

The program performs measurements not only one, when the screen is turned on, but as long as there is an interaction between the device and the user. Time between measurements differs. If the user is defined as an intruder, there are two possible versions of events. Firstly, he cannot open a protected application, and then the measurement will be carried out over a certain period of time *T*. Secondly, as a consequence of a failed attempt to open a protected application because of an invalid password, it will deny access to all Protected Applications for a period of time. In this case, is not necessary to take measurements while applications are blocked. The following data collection occurs after a timeout. If the algorithm assigned the user as a legitimate user, then the next measurement will occur after a longer period than *T*.

Too frequent sensor data collection can lead to an overflow of device memory and that does not satisfy any potential users. On the other hand, collecting data too rarely, there is a risk to miss the measurement corresponding to the attacker, in this case the owner's sensitive data will be compromised. As you can see, you need to choose the right time *T*.

Whatever the chosen time *T*, the more days pass from the date of installation of the application, the more this application will take up memory. To avoid such problems, it is necessary to keep the size of the training sample. Defined as belonging to the legitimate user each new entry replaces one of the older properties already owned by the sample. Substitution must be such that the border of user's objects area will not decrease. In other words, the samples of legitimate emissions should not be excluded. Such emissions may correspond to, for example, the new geolocation, where do he visits from time to time.

IV. Machine learning algorithms

It is necessary to analyze a wide variety of data and information to identify an intruder. This problem can be solved by methods of machine learning which give a very accurate result without time and intellectual costs for the user. Machine learning is programming computers to optimize a performance criterion using example data or past experience [10]. That past experience allows to solve the required tasks in a short time.

The best-known machine learning classes are supervised learning and unsupervised learning. The main difference between supervised and unsupervised learning is that the training sample in the first case consists of inputs and outputs, and in the second case there are no outputs or not enough of them. Both classes include a plurality of algorithms, which can be divided according to the type of their tasks. The most famous problem are the problem of classification, regression, clustering and others. In the work [11] more details of some problems are presented.

During the authentication, it is necessary to check whether the person holding the device is the legitimate user. This problem could be attributed to the problem of classification, but in this case, it is necessary to know the exact number of object classes and the number of objects in each class should be sufficient. The class corresponding to the measurements of a legitimate user, will have plenty of measurement, and the class that represents intruder data is empty or very small. This task description corresponds exactly to the problem of anomaly detection [12].

Anomalies are those data which do not correspond to a predetermined normal behavior. There are no examples of anomalies or a limited number of them in the training set and we cannot determine where they exactly are. Therefore, the problem of anomaly detection belongs to unsupervised learning. Each sample object is described by a set of attributes. It is their nature who determines the applicability of a particular method of machine learning to the data. Frequently, there are objects dependent on each other, the presence or absence of such interconnection play a role in the workability of the chosen method.

There are two approaches to the detection of anomalies. Basic methods are based on the restoration of the distribution density, also this problem can be reduced to a classification problem. In a probabilistic approach to the detection of anomalies it is considered that anomaly is an object that has been received from a distribution other than the one with which the training sample was generated. If you find this distribution, it will be possible to evaluate the probability of

belonging this object to distribution. If the probability is very low, then most likely, the object is an anomaly.

It is generally agreed that, there are two approaches to the restoration of density: parametric and non-parametric. A nonparametric approach differs from a parametric as it tries to restore the distribution only of the data, without using any of the family of distributions. To use the parametric approach, it is necessary to know which of families of distributions describes the training sample data, only in this case it can be applied. It often happens that the objects belong to two or more distributions. Then, recovery of distribution mixtures method is used. When exploring the new object, the probability that it belongs to the reconstructed the distribution density is calculated. Then, the obtained value is compared with a threshold value, if it is less than an anomaly. The threshold value can be selected from a priori considerations, either for known anomalies.

Let there is some probability distribution on all objects that can be obtained. This distribution is considered a parametric because each X dependents on some parameter. One of the most well-known parametric distribution is a normal distribution. It is necessary to define the parameters, to try to recover the sample using a normal distribution. Parameters should be chosen so that the probability that the objects of the training samples belong to this distribution was a maximum. Then, objects that are not related to the sample will have low probability. The maximum likelihood method is easy to handle this task [13]. It is trying to pick up a distribution from a parametric family thus, the objects of the training sample were the most likely. But working with the likelihood is inconvenient, its logarithm is usually taken. Then, the sum of the logarithms is maximized.

Fisher's linear discriminant refers to statistical and machine learning methods that are used to find linear combinations of features that best separate two or more classes of objects or events. For each sample object or event with the known Y class a set of attributes X is considered. A set of such samples have training sample. The challenge is to build a good prognosis for each object in class having only the observation X.

It is expected that the density of probability distribution is normal for both classes. In this case, the objects belong to the second class if the likelihood ratio below a certain threshold, and to first unless it is higher. It happens that the training sample is generated from two normal distributions with different centers, but identical covariance matrices. It is not possible to describe such a sample with normal distribution. For such a case, mixture distributions model is suitable. The mixture is called a distribution, which is represented as a weighted sum of the other distributions.

Distributions of the incoming mixture called components and they are generally parametric distributions. To restore the distribution density of the mixture, EM-algorithm is used. EM-algorithm is an algorithm used in mathematical statistics to find estimating of maximum likelihood of the probabilistic model parameters, when the model depends on several hidden variables. Each iteration consists of two steps.

In the E-step the expected value of the likelihood function is calculated, and the hidden variables are considered to be observed. In the M-step the maximum likelihood estimation is calculated. Thus, the expected likelihood, which was calculated on the E-step increases. This value is then used for the E-step in the next iteration. The algorithm is performed until convergence.

An example of a nonparametric recovery of density is the method of Parzen window [14]. Nonparametric estimation of the density is generalized well in the multi-dimensional case, and the difference between the points is replaced by the metric, the normalization constant is introduced to normalize the density. This approach has one big problem. The number of objects required to restore density increases exponentially with the dimension of the space. Therefore, in practice, this approach is used in spaces with not very high dimension.

Some properties of algorithm inputs can belong to one of the families of distributions, so in the further work we cannot exclude the non-parametric methods for the recovery of density distribution.

The task of searching the anomalies may be associated with the classification task. It is believed that all objects of the training sample are normal, while coordinate origin is an anomaly. Now, this problem can be solved as a problem of classification. We use a linear way, choosing hyperplane to split objects so that it gives the maximum size of the gap in order to reduce the possibility of re-education. Support Vector Machines [15] is used for that.

The assumption made in the previous method that 0 is anomalous object is very strange. If the sample is centered near the origin of coordinates, this method cannot give the correct answer. Therefore, a support vector machine with a linear kernel never used. So, the scalar product of vectors is replaced by the kernel K. Thus, the separating hyperplane is constructed in the space of a higher dimension. The RBF-kernel is commonly used. So, a second class describing anomalies is created and binary classification methods can be applied.

Let us consider some binary classification algorithms and evaluate their applicability to the user authentication problem.

A very simple algorithm is Naive Bayes. Naive Bayes classifier is a special case Bayesian classifier [16] based on the additional assumption that the objects of X described N statistically independent features. The assumption of independence of features greatly simplifies the task because it is easier to estimate the N one-dimensional density than one the N-dimensional.

Unfortunately, in practice, this assumption is very rarely performed, so Naive Bayes classifier is used mostly for comparison with other models of algorithms as a primitive model, or as part of an algorithmic composition. The training

set consists of objects provided by smartphone sensors that correspond to user behavior. Such measures as the area of contact of a finger with the touch screen and the speed of movement on it will depend on each other. Therefore, Naive Bayes is not applicable to the problem of authentication, which uses these biometric data as properties of the input of machine learning algorithm.

The easiest way to assess the quality of the algorithm is to use the left-off sample. The sample is divided into two parts, one of which will serve as a training, and the second will be a test set to evaluate the quality of the method on it. In this case, it is necessary to determine the proportion of the partition sample. If a test sample is too small, the quality assessment algorithm is unreliable. Otherwise, if a test sample is too large, the learning sample is small and the algorithm quality will not be high.

Nearest neighbor method is the simplest metric classifier based on the evaluation of the objects similarity [17]. The classified object belongs to the class to which belong nearest from training sample objects. The hypothesis of compactness claims that close objects usually belong to the same class. It is necessary to formalize the definition of nearness. To do this, a distance function that places the pair of objects in compliance with non-negative number is selected. You can enter a requirement that such a function has a metric (symmetrical, and follow the rules of the triangle), but this condition is not necessary. To include an input X to any class, you need to arrange the sample objects ascending distances to X, set the weight of each "neighbor" and its contribution in classification. Nearest neighbor method is the simplest, but it is unstable if emissions occur. Therefore, the probability that X will be assigned to the wrong class increases. It is necessary to check its performance on a problem arised in this article.

Certainly, the problem of binary classification can be solved with the help of neural networks [18]. Consider an elementary perceptron. In reference [19] the perceptron is a feedforward network containing a retina that is used only for data acquisition and which has fixed-weighted connections with the first neuron layer (input layer). The fixed-weight layer is followed by at least one trainable weight layer. One neuron layer is completely linked with the following layer. The first layer of the perceptron consists of the input neurons defined above. Elementary perceptron is a simple perceptron, in which all the elements are simple, i.e. they realize the threshold function.

When a signal applies to the perceptron input, some sensor elements are excited. Information about S-element enters the associative elements (each of which corresponds to the few S-elements), if the signals received on the A-element, and it is sufficient for the element to become excited, it transmits a signal to the responsive element. Each signal supplied to the A-element multiplied by the weight value from the matrix V and is summed to other signals within a single R-element. If the sum exceeds the threshold then perceptron outputs "+1", else "-1". A function that allows to implement these calculations is called the threshold.

Single-layer perceptron is a model of perceptron in which the input elements are connected directly to the output with the help of the weighting system. In this case, each of A-element corresponds with a single S-element, all S-A references have weight 1, and the threshold of A-elements is zero. Such neural network is called a linear classifier.

Radial basis function network is an artificial neural network, in which the radial basis function (RBF) is the activation function. In the radial basis function, there are three features. There is only one hidden layer, hidden layer neurons only have a non-linear activation function, the weight of the synaptic connections of the input and hidden layers are equal to one. The output of such a network is a linear combination of RBF neuron inputs and parameters.

The single-layer perceptron and RBF network can be used for binary classification.

Consider a more complex model. Examples of such models can serve as decision trees [20] and random forest [21]. Typically, decision trees are binary trees. In each interior point condition is stored, and each of leaves is a recorded prognosis. The chosen conditions are extremely simple, compare input with the value of a certain attribute threshold. The result (or forecast) in the case of classification problem is belonging to a certain class.

It should be considered when building decision trees that the minimization of errors leads to relearning of the tree. We can build minimum tree describing the sample so that the error on the training data is zero. A tree is now retrained, so it will not ignore the emissions and build a very complicated separating surface. It follows that we must be able to build decision trees correctly. To do this, there are many different techniques [22].

In fact, the trees themselves are almost not used today, they are only needed for the construction composition and combining a large number of trees in a single algorithm. To avoid overfitting decision trees are combined in a composition and become one not re-trained algorithm.

One of the best ways to combine decision trees in composition is random forest. The process of constructing decision trees is a greedy algorithm that works before the stop criterion.

In addition to choosing a method of machine learning, which will solve the problem, you need to determine how to create a high-quality algorithm. The partition should not be made so that all objects with similar features are in the same part of the sample. With this partitioning algorithm, will give the wrong answers on the test values. To eliminate this problem, you can use the this approach. Build K various partitions of a sample into 2 parts, for each partition calculate the assessment of the quality, average the quality assessment on all partitions. The resulting average value will be used as a final assessment.

Cross-validation is an improvement of left-off sample. The differences lie in the first step. The entire sample is divided

into K blocks of approximately equal size. Then, each of the blocks is used as a test, while the rest are used as the training set. Using many blocks, the quality assessment obtained is reliable, but biased, using a small number of blocks, results are unreliable, but unbiased. The number is chosen based on the conditions of a specific task. It should be understood that the greater the number of blocks is, the more time will be needed have to train the algorithm. It is better to choose a small value K to a large sample.

V. Conclusion

In this study an architecture of an application that provides continuous user authentication is proposed. Machine learning algorithms were considered and the applicable ones were chosen.

The following research will be held in the purpose of developing the proposed application and estimation of its performance.

Acknowledgements

This work was supported by Competitiveness Growth Program of the Federal Autonomous Educational Institution of Higher Professional Education National Research Nuclear University MEPhI (Moscow Engineering Physics Institute).

References

[1] E. Owusu, J. Han, S. Das, A. Perrig, and J. Zhang, "Accessory: password inference using accelerometers on smartphones", *HotMobile'12*, 2012, P. 9:1–9:6.

[2] N. Zheng, K. Bai, H. Huang, and H. Wang. "You Are How You Touch: User Verification on Smartphones via Tapping Behaviors", *Proceedings of the 2014 IEEE 22nd International Conference on Network Protocols*, 2014, P. 221-232.

[3] M. Frank, R. Biedert, E. Ma, I. Martinovic, and D. Song "Touchalytics: On the Applicability of Touchscreen Input as a Behavioral Biometric for Continuous Authentication", *IEEE Transactions on Information Forensics and Security, Vol. 8.,* 2013, P.136-148.

[4] H. Crawford "Keystroke dynamics: Characteristics and opportunities" *Privacy Security and Trust (PST), 2010 Eighth Annual International Conference*, 2010, P. 205–212.

[5] A. Alzubaidi, and J. Kalita, "Authentication of Smartphone Users Using Behavioral Biometrics", *Journal of IEEE Communications Surveys and Tutorials,* Vol. 18, 2016, P. 1998-2026.

[6] M. Derawi, and P. Bours, "Gait and activity recognition using commercial phones", *Computers & Security,* Vol. 39, 2013, P. 137-144.

[7] M. Jakobsson, E. Shi, Ph. Golle, and R. Chow "Implicit Authentication for Mobile Devices" Proceedings *of the 4th USENIX conference on Hot topics in security*, 2009, P. 9-19.

[8] J. Zhu, P. Wu, X. Wang, and J. Zhang, "Sensec: Mobile security through passive sensing", *International Conference on Computing, Networking and Communications (ICNC)*, 2013, P. 1128–1133.

[9] L. Li, X. Zhao, and G. Xue "Unobservable re-authentication for smartphones", *In NDSS*, 2013.

[10] E. Alpadin, *Introduction to Machine Learning.* London: The MIT Press, 2010, 579 p.

[11] R. Rojas, *Neural Networks: A Systematic Introduction.* Berlin: Springer-Verlag, 1996, 502 p.

[12] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly Detection: A Survey", *ACM Computing Surveys*, Sep.2009, P. 1-72.

[13] I. J. Myung "Tutorial on maximum likelihood estimation", *Journal of Mathematical Psychology 47*, 2003, P. 90-100.

[14] L. Lan, H. Shi, Z. Wang, and S. Vucetic, "An Active Learning Algorithm Based on Parzen Window Classification", *JMLR: Workshop and Conference Proceedings 16,* 2011, P. 99-112.

[15] Y. Tang. "Deep Learning using Linear Support Vector Machines", *In Workshop on Representational Learning, ICML,* 2013.

[16] S. Raschka *Naive Bayes and Text Classification I: Introduction and Theory.* Ithaca: Cornell university library, 2014.

[17] O. Sutton, "Introduction to k Nearest Neighbor Classification and Condensed Nearest Neighbour Data Reduction", February, 2012.

[18] K. Gurney *An introduction to neural networks*. London: UCL Press, 1997, 317 p.

[19] D. Kriesel *A Brief Introduction to Neural Networks,* Bonn: University of Bonn, 2005, 244 p.

[20] S. B. Kotsiantis "Supervised Machine Learning: A Review of Classification Techniques", *Informatica 31,* 2007, P. 249-268.

[21] L. Breiman "Random forest", *Machine learning*, 45(1), 2001, P. 5-32.

[22] A. Kak *DECISION TREES: How to Construct Them and How to Use Them for Classifying New Data,* An RVL Tutorial Presentation, 2016, 127 p.