Comparative Analysis of using Recurrent Autoencoders for User Biometric Verification with Wearable Accelerometer

Mariia Havrylovych [1[0000-0002-9797-2863]] , Valeriiy Danylov [2[0000-0003-3389-3661]]
Aleksandr Gozhyj[3[0000-0002-3517-580X]]

[1] National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute",
Peremohy avenue 37-A**,** Kyiv, 03056, Ukraine
mariia.havrylovych@gmail.com
[2] National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute",
Peremohy avenue 37-A**,** Kyiv, 03056, Ukraine
danilov1950@ukr.net
[3]Petro Mohyla Black Sea National University, Desantnykiv st. 68 Mykolaiv, 54000, Ukraine
alex.gozhyj@gmail.com

Abstract. A comparative analysis use of recurrent auto-encoders as a component of the proposed automated decision support system for biometric verification of the behavioral type user from the indicators of the chest accelerometer sensor is carried out. The purpose of the proposed system is to perform continuous, implicit user verification based on the accelerometer, to improve the security usage of application or device, as well as improve user interaction with the device. The goal of research is to build a continuous-based user biometric verification system based on accelerometer data using unsupervised deep learning recurrent algorithms. Auto-encoders enable one-class classification, i.e. unsupervised learning, as well as additional layers in the network architecture allow to automatically generate features and encode a data input into a feature vector, which greatly facilitates data processing and makes the system more automated. The purpose of the model is to define the boundaries of the positive class, namely to distinguish a specific user from others, which in fact is to solve the problem of detecting anomalies. The comparative analysis compared three types of recurrent auto-encoders with recurrent units of long-short term memory type with two methods of classical machine learning (one-class support vector machines and “isolation” forest) which required manual feature generation. Proof of concepts of using recurrent autoencoders for biometric verification was implemented and tested on the open-source dataset. Usage of recurrent autoencoders for user behavioral-based verification has shown robust and high accuracy results and the ability of implementing such algorithms in modern security systems.

Keywords: Biometric Verification, Recurrent Auto-encoder, Anomaly Detection, Variational Auto-encoder.

1 Introduction

Nowadays our phones and other big variety of electronic devices become a great part of our life. We have a lot of information on our phones, even such important info like the number of credit cards, credentials from some payment service, etc. Also, now it is very important to make analytics on your devices, for further development of different data-driven decisions, which will improve the level of user satisfaction from using the app and provide a smart decision-making system on the user device. So, it is very important to provide accurate user identification.

The problem of user verification, that this is impossible to create a supervised task because we can't compare one particular user with all other users. So, this becomes the unsupervised task and the aim is to detect the boundaries of one good class and all the samples that don't belong to it, will be considered as anomalies. In the context of user verification, this will be other users. The data from the accelerometer is in a time-series format. The objective is to solve the anomaly detection problem on time series data, which is the task of higher complexity.

Reference to [1] we have 3 types of user personalization system: knowledge-based (password), possession-based (token, smart card) and biometric-based, which divides in physiologic (face-based, retina-based, fingerprint-based) and behavioral (gait recognition, keystroke dynamics, mouse movements, signature recognition). Gait recognition means to find the pattern of how a man walks, stands, works at a computer and is detected with an accelerometer. Machine learning and especially deep learning approaches are widely used for solving biometric-based user personalization problems [2]. In our work, we research the behavioral-based personalization system, which will define behavioral patterns in different types of user activity (walking, standing, working at the computer) in data collected with a chest accelerometer. The advantage of the autoencoder-based approach that it is enough to have data only from one positive class and we are solving one-class classification task. In our work, we propose using a different type of autoencoders for biometric verification.

An object of the study is the user biometric verification based on behavioral patterns. The accurate and speed behavioral-based verification allows making it continuous and implicit, which is very important to create secure and protected services for different purposes.

The subject of study is unsupervised machine and deep learning-based algorithms for one-class classification.

The purpose of the work is the comparative analysis of using different unsupervised deep learning autoencoder-based approaches for solving biometric verification problem which is, in fact, an anomaly detection problem.

2 Problem statement

Given the set of time series windows of length *l* and overlapping percent *p* with particular activity a { x_a^i }- *i*-th data instance with activity *a*, where i =1,2,...N (N – number of instances in train set).

The problem of user verification can be formulated as finding such T_{opt}, which is optimal for next target functions:

$$\{\sum_{i=1}^{N}\delta(E(x_i) < T) \to \max, \sum_{j=1}^{M}\delta(E(z_j) < T) \to \min\},$$

where T - error threshold, which detects abnormality of given data instance; T_{opt} - optimal threshold; δ - function which get the logical expression and returns 1 if it is true, otherwise 0; *E(x)* - model error on given *x* data instance; *M* - number of anomaly data instances; z_j - *j*-th anomaly data instance.

3 Literature review

In [3] research authors propose a sophisticated continuous-based authentication system for implicit verification on smartphones based on motions patterns using autoencoders (with 1,3 and 5 layers). They use simple autoencoders, but pay attention to developing distributed cloud architecture to make possible expensive computations for smartphones on the cloud and therefore make it faster and computationally efficient.

Also, it is possible to use autoencoders not for authentication but for the inverse problem - anonymization, to prevent deanonymization when sending data to untrusted resources [4]. In [4] proposing specific anonymization function, which based on adding regularization to the loss function. Deep learning approach to user verification using Deep Clockwork RNN proposed in [5]. The recurrent approach is important when using sequential and time-series data, such as accelerometer sensor data, because the previous signals have an impact on the future, but in [5] authors didn't solve a one-class classification problem.

Not only accelerometer data can be used for biometric identification, but also ECG or EEG data. In [6] authors also use deep autoencoders for feature learning part of the personal identification system, but with ECG data. This autoencoder approach may be helpful not only for user personalization but also for finding anomalies in user health indicators, and this is extremely important because, with historical data, you can make some analytics, and make some more sophisticated inference about user-health overall, and of course it may help in some critical situation. In [7] authors build biometric authentication method combining cryptography techniques with biometrics (EEG), like Bose-Chaudhuri-Hocquenghem (BCH) codes. But we need to have the database of users to be able perform the authentication for proper user matching, which is not always possible. Also, ECG and especially EEG is hard to record for now continuously, while movements and activities with accelerometer are easy to access.

The idea of implicit authentication based on behavioral patterns such as a combination of all data sources which your smartphone can propose, like application usage, phone call patterns, etc. proposed in [8]. It is useful for smartphones, but for example, when you have some fitness device like bracelets, you will not have access to such variety of user data, so it is important to be able to provide ability of the user verification system to make authentication based only on sensor data.

Another set of algorithms that can be used as anomaly detection method are artificial immune systems (method of positive and negative clonal selection for example), and in [9] there is an artificial immune system approach for user personalization based on touch-based behavioral patterns.

So, for working with sensor time-series data, we need to use recurrent architectures but as well we want to use unsupervised algorithms both for feature engineering and for detecting non-self users. In this work, we combine all these parts as one model to check the ability of such algorithms to solve such complex, multistage problems.

4 Materials and methods

For biometric user verification were used autoencoders.

Here we look deeper into autoencoder internals.

Autoencoder architecture contains two main parts: encoder and decoder.

Autoencoders have learned to create a reconstruction of the original input. The goal is to minimize reconstruction error:

$$E = \sqrt{\sum_{i=1}^{n} \left\| x_i - d_\varphi(e_\theta(x_i)) \right\|}, \tag{1}$$

Where $x_1 ... x_n$ is data rows, d is decoder and e is encoder with some parameters φ and θ respectively.

Encoder encode input in some lower dimensional or higher-dimensional space. It cannot be just copied from input to output, because we put some constraints, like lower dimension of inner layers and for being able for the decoder to recreate the output, encoder have to find and extract some meaningful patterns and features. Decoder’s purpose is to recreate the sample from an encoded example.

The autoencoder training algorithm is updating parameters of decoder and encoder using gradient descent-based algorithms to minimize (1) [10]. After training, we set some threshold ε and compute reconstruction error for input data and if reconstruction error for some data point is higher than the threshold, we assume that this data point is an anomaly.

The threshold setting is not strictly defined and leaves up to the researcher. So, this part can be customized, based on the particular qualities of the problem the researcher needs to solve.

There are multiple types of autoencoders:

1. Undercomplete (when the dimension of inner layers is smaller than input), encoder solve dimensionality reduction problem.
2. Sparse (when the dimension of inner layers is bigger than input), adding some sparsity penalty to reconstruction error.
3. Denoising (adding some noise to input and put it into reconstruction error). It minimizes not (1) but

$$\sqrt{\sum_{i=1}^{n} \left\| x_i - d_{\varphi}(e_{\theta}(\overline{x_i})) \right\|}, \qquad (2)$$

Where $\overline{x_i}$ is x with some noise.

4. Contractive (add the penalty Ω(h) is the squared Frobenius norm (sum of squared elements) of the Jacobian matrix of partial derivatives associated with the encoder function)[10]:

$$\Omega(h) = \lambda \left\| \frac{\partial f(x)}{\partial x} \right\| .$$

5. Variational autoencoders (VAE) which optimize reconstruction probability. VAE is a generative model, which tries to reconstruct parameters of the probability distribution of output.

Variational autoencoders are directed probabilistic graphical model. Variational autoencoders don’t minimize reconstruction error but optimize reconstruction probability.

It is absolutely different approach, we still have the encoder and decoder, but decoder doesn’t decode, but sample data examples from some probabilistic distribution with some parameters and encoder maps data samples into latent probabilistic space, so in core of loss function for variational autoencoder we have Kullback–Leibler divergence, which shows the difference between different probabilistic distributions [11].

In [12] authors research is that possible to combine recurrent neural network and variational autoencoder.

So based on previous consideration the purpose is that for user personalization problem solving based on chest accelerometer sequential data it is important to use recurrent architectures to think about previous values, and because of high complexity of solving classification problem of distinguishing users, the best approach is to define boundaries of “self” class (the user in this case), and for this we need unsupervised machine learning techniques.

Using recurrency and unsupervised learning together the answer is recurrent autoencoders.

Let's propose a continuous verification system (fig. 1).

6. Collecting accelerometer data (or other sensor data, that describe motion patterns) for different activities type.
7. Recognize human activities and divide data into parts by activities.
8. Train models and fine-tune parameters.
9. Choose the most appropriate model for a particular user based on a low false-positive rate.

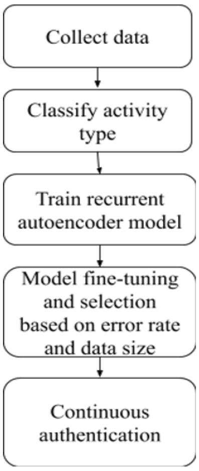

Collect data
Classify activity type
Train recurrent autoencoder model
Model fine-tuning and selection based on error rate and data size
Continuous authentication

Fig. 1. Schema of the user personalization system

5 EXPERIMENTS

Dataset was taken from UCI repository - archive of public datasets for machine learning purposes.

The dataset contains accelerometer data from 15 volunteers. Data contains from x, y, z axis values [14]. The sampling frequency of the accelerometer is 52 Hz, so we have 52 rows of values each second. Also, data contains 7 activities labels (standing up, walking and going up and downstairs, standing, walking, going up and downstairs, walking and talking with someone, talking while standing). As in [15] user verification goes after solving human activity recognition (HAR) tasks on walking patterns, we also use the user verification algorithm only on some particular motion pattern. So, actually we detect in which way some person walking or walking and talking with someone etc. The most rows from volunteers were with working at a computer label, so we decide to train models with working at computer activity.

For deep learning models, we split data in overlapping on 50 percent windows with a length of 52.

The number of train and test samples shown in table 1.

We build autoencoders with python on Keras library and Tensorflow backend [16].

We compare three types of autoencoders:

- variational long-short term memory (LSTM) autoencoder;
- contractive LSTM autoencoder;
- undercomplete LSTM autoencoder.

As loss was used mean absolute error. Train in 10 epochs with 32 batch size and Adam optimizer.

The threshold formula:

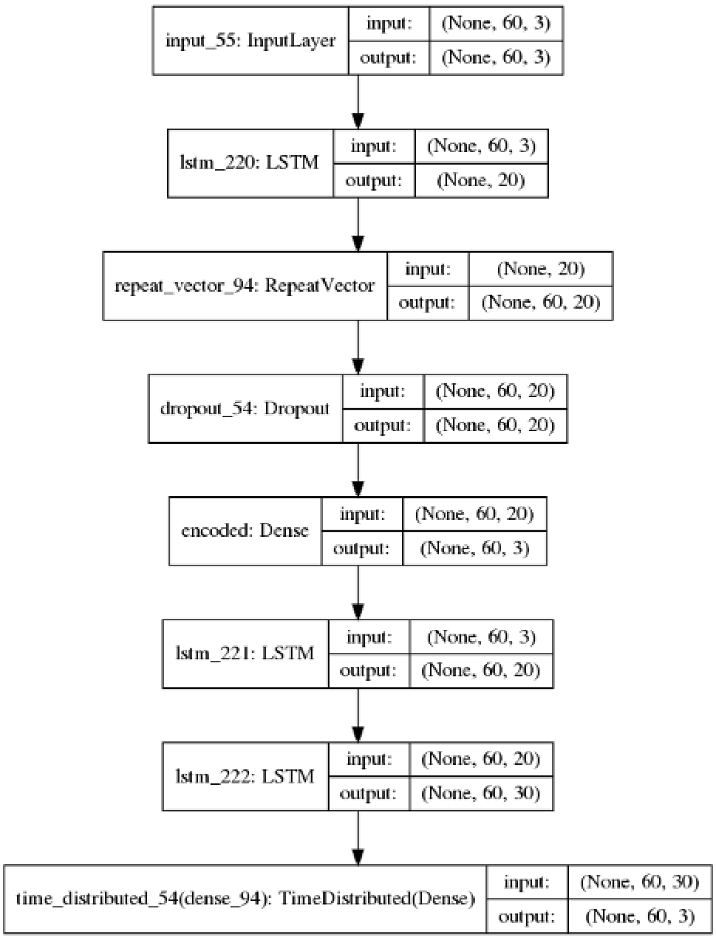
$$T = \sum_{i=1}^{N} MAE_i \big/ n + std(MAE_i)$$

where MAE is mean absolute error between a sample and predicted sample, std - standard deviation and n is the number of samples in the train dataset.

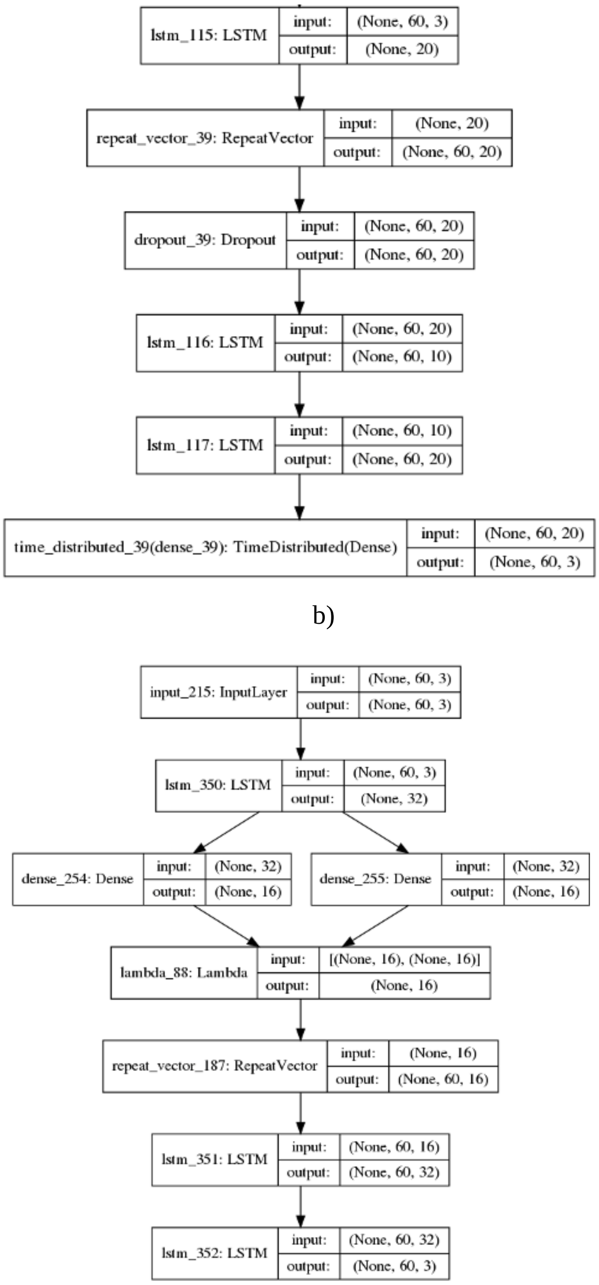
The architecture of autoencoders is shown below at fig. 2.

As model evaluation metric “recall” for positive (“self”) class were chosen and also area under curve (AUC) value for correctly compare models with each other, not depending on classification threshold. Also equal error rate was calculated.

Autoencoder-based models were compared with one-class support vector machines (SVM) and isolation forest algorithms. For one-class SVM and isolation forest, not raw data, but different types of features were extracted (in time and frequency domain). The examples of features were taken as in [17].



a)



b)

c)

Fig. 2. Architecture of a) contractive autoencoder, b) variational autoencoder, c) undercomplete autoencoder

6 RESULTS

Positive class recall score for every user and comparing with one-class SVM and isolation forest are shown in table 1-3 below.

Also because of not very high difference in recall score in autoencoder models, we compare them with area under curve value. ROC curve and AUC value is shown in fig. 3 and fig. 4.

Table 1. Number of train and test samples for 1-15 users

User	1	2	3	4	5	6	7	8
Train	866	1136	1072	812	797	1133	842	1133
Test	428	561	529	400	393	559	416	559
User	9	10	11	12	13	14	15	
Train	929	1134	1394	1254	470	1361	1328	
Test	458	559	688	619	232	671	655	

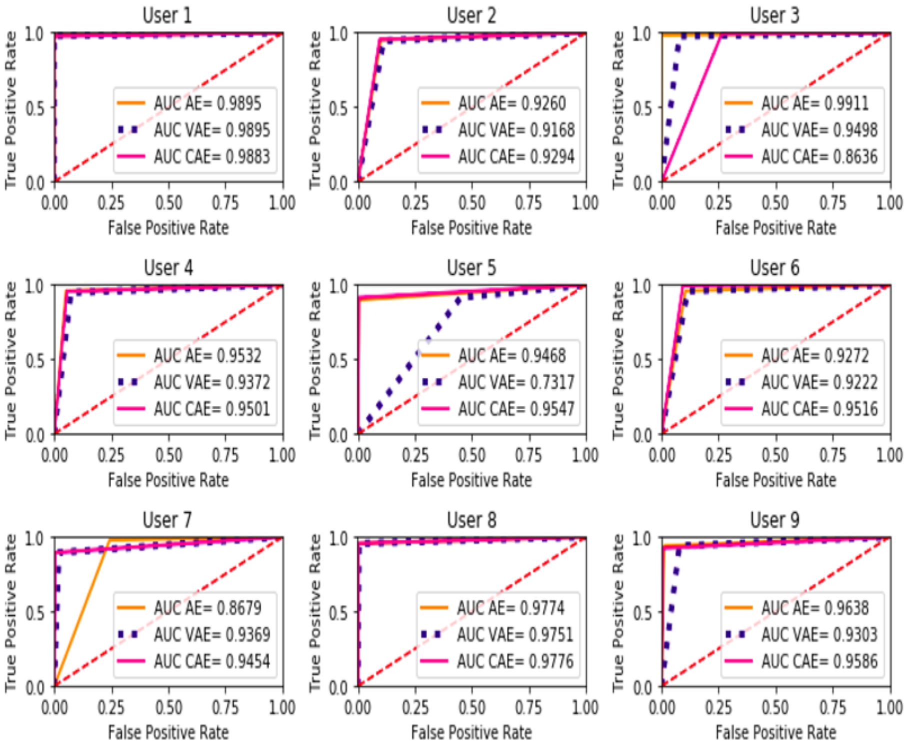


Fig. 3. ROC curve and AUC score for autoencoder-based models (users 1-9)

Table 2. “Recall” score for the positive class

User	LSTM AE	LSTM VAE	LSTM CAE	SVM	IF
User1	0.99	0.99	0.99	0.63	0.74
User2	0.99	0.99	0.66	0.42	0.73
User3	0.62	0.96	0.86	0.67	0.74
User4	0.44	0.58	0.66	0.27	0.74
User5	0.94	0.94	0.95	0.67	0.78
User6	0.87	0.91	0.91	0.49	0.74
User7	0.91	0.92	0.90	0.65	0.65
User8	0.97	0.97	0.97	0.39	0.83
User9	0.50	0.55	0.56	0.27	0.76
User10	0.94	0.97	0.97	0.29	0.83
User11	0.98	0.98	0.98	0.68	0.77
User12	0.58	0.41	0.46	0.40	0.84
User13	0.77	0.27	0.59	0.20	0.88
User14	0.76	0.73	0.82	0.32	0.72
User15	0.96	0.98	0.98	0.66	0.79
Average	0.814	0.81	0.82	0.47	0.77

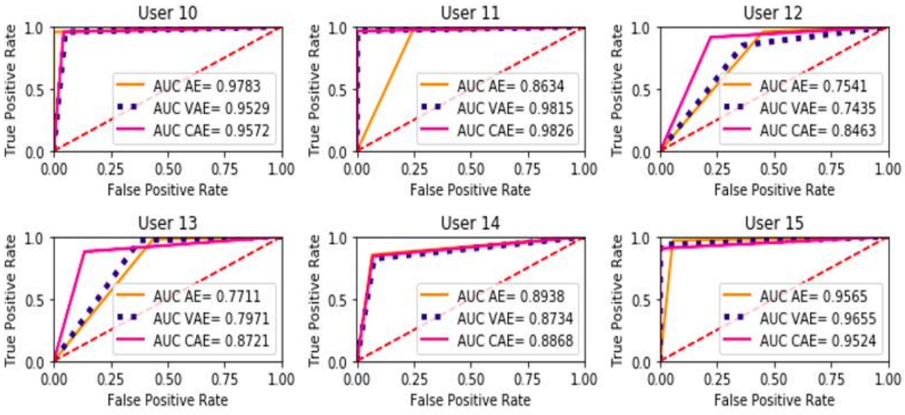


Fig. 4. ROC curve and AUC score for autoencoder-based models (users 10-15)

Table 3. Average equal error rate (EER) for every autoencoder-based model

	LSTM AE	LSTM VAE	LSTM CAE
EER	0.125	0.111	0.106

In table 2 we can see that the best result showed recurrent contractional autoencoder with 0.82 average recall score for positive class.

Based on EER value in table 3 contractive and variational autoencoder show almost the same performance.

On the fig. 3 and fig. 4 we can see ROC curves and AUC values for every type of autoencoders that was used in experiment. AUC value allows us to compare the models in general, while “recall” is calculated on some fixed threshold. Thus, AUC metric is better for model comparison, while “recall” score should be considered more as final key performance indicator (KPI) for specific business usage. Based on the results, AUC value is different depending on user, so we can say that the model should be adaptively selected based on performance, error rate and amount of data for particular user. However, contractive autoencoder is still a leader, as it has highest result in 8 users from 15, the second is classic autoencoder.

7 Discussion

Following from previous results as we can see recurrent autoencoders show robust results with a high value of chosen metric for evaluation. Also, depending on data size and particular user result can be different. For every user, the best way will be to choose the model with the highest recall and AUC value. Comparing to a one-class SVM and isolation forest where feature where manually engineered, automatically feature extraction as part of the autoencoder model doesn’t underperform, which is important for ubiquitous usage, its versatility, and portability for different types of sensors and data sources.

But overall most robust results were shown by recurrent contractive autoencoder.

Also, should be considered the data separability degradation with increasing the numbers of user. In our case we have only 15 users, which is a pretty small number compare to the real-world tasks. In this case, we have to understand how properly train and evaluate the model.

For example, we can cluster users in different group and evaluate the model only inside specific group of most similar user. Thus, we will choose the model which distinguish “self” user from others, even the most similar one.

We need to consider possible issues, that may appear in using such systems in real world problem, as: a big number of similar users, noisiness of accelerometer signal, high model latency, etc. Another loss should be considered as well in order to customize the problem for specific biometric purposes.

Our research shows that using deep unsupervised learning can be successfully used as part of biometric continuous personalization decision support system.

8 Conclusions

Here we compare different autoencoders for biometric user personalization based on motion patterns. The purpose was to define the boundaries of positive class to distinguish particular users from the rest. Undercomplete LSTM autoencoder, LSTM variational autoencoder, and contractive LSTM autoencoder were compared with one-class SVM and isolation forest. Recurrent autoencoders show robust and high accuracy results. The advantages of autoencoders are the unsupervised approach and automatic feature extraction, engineering, and selection, the disadvantage is their requirements for computational resources.

The contractive autoencoder show the best results with 0.82 recall score for positive class and highest AUC value for 8 users from 15. Contractive autoencoder show the lowest equal error rate as well (0.106). The other type of autoencoders, as classic autoencoder and variational show the good result with 0.81 recall score for positive class. Compare with classic machine learning algorithms as isolation forest and one-class SVM with 0.77 and 0.47 recall score respectively autoencoders show much better results.

In the future, we can extend the list of autoencoder models, and use more sophisticated cascades of autoencoders, like stacking autoencoders of different types in one big model or adding convolutional and subsampling layers before encoder to improve feature extracting part of the model, but there may be an issue with computational efficiency. Also, there has to be research about optimization using deep learning models in different decision support systems. As well there is a need to propose the more complex evaluation metric for such types of models.

References

1. Mahadi, N.A., Mohamed, M.A., Mohamad, A.I., Makhtar, M., Kadir, M.F.A., Mamat, M.: A Survey of Machine Learning Techniques for Behavioral-Based Biometric User Authentication. Recent Advances in Cryptography and Network Security, pp.43-54 (2018).
2. Sundararajan, K., Woodard, D.L: Deep Learning for Biometrics. ACM Computing Surveys (CSUR) 51, pp. 1 – 34 (2018).
3. Centeno, M.P., Moorsel, A.V., Castruccio, S.: Smartphone Continuous Authentication Using Deep Learning Autoencoders. 2017 15th Annual Conference on Privacy, Security and Trust (PST), pp. 147-1478 (2017).
4. Malekzadeh, M., Clegg, R.G., Cavallaro, A., Haddadi, H.: Mobile sensor data anonymization. Proceedings of the International Conference on Internet of Things Design and Implementation, pp.49-58 (2019).
5. Neverova, N., Wolf, C., Lacey, G., Fridman, L., et all: Learning Human Identity From Motion Patterns. IEEE Access. 4, pp. 1810–1820 (2016).
6. Eduardo, A., Aidos, H., Fred, A.: ECG-based Biometrics using a Deep Autoencoder for Feature Learning - An Empirical Study on Transferability. Proceedings of the 6th International Conference on Pattern Recognition Applications and Methods, pp. 463-470. (2017).
7. Damasevicius, R., Maskeliunas, R., Kazanavicius, E., Woźniak, M.: Combining Cryptography with EEG Biometrics. Computational Intelligence and Neuroscience. pp. 1-11 (2018).
8. Shi, E., Niu, Y., Jakobsson, M., Chow, R.: Implicit Authentication through Learning User Behavior. Lecture Notes in Computer Science Information Security, pp. 99–113 (2011).
9. Aljohani, N., Shelton, J., Roy, K.: Continuous Authentication on Smartphones Using An Artificial Immune System. Proceedings of the 28th Modern Artificial Intelligence and Cognitive Science Conference 2017, Fort Wayne, IN, USA, pp. 171–174 (2017).
10. An, J., Cho, S.: Variational Autoencoder based Anomaly Detection using Reconstruction Probability (2015).
11. Goodfellow, I., Bengio, Y., Courville, A.: Deep learning. MIT Press, Cambridge, MA (2017).
12. Fabius, O., Amersfoort, J.R.van, Kingma, D.P.: Variational Recurrent Auto-Encoders. ICLR 2014, CoRR (2014).

13. Dua, D. and Graff, C. UCI Machine Learning Repository. Irvine, CA: University of California, School of Information and Computer Science (2019).
14. UCI machine learning repository, https://archive.ics.uci.edu/ml/datasets/Activity+Recognition+from+Single+Chest-Mounted+Accelerometer, last accessed 2020/15/10.
15. Casale, P., Pujol, O., Radeva, P.: Personalization and user verification in wearable systems using biometric walking patterns. Personal and Ubiquitous Computing. 16, 563–580 (2011).
16. Chollet, F.: The Keras Blog, https://blog.keras.io/building-autoencoders-in-keras.html, last accessed 2020/15/10
17. Thingom, B. S., Rajsekhar, K. N., Narsimhadhan, A. V.: Person Recognition using Smartphones' Accelerometer Data. CoRR abs/1711.04689 (2017).
18. Nguyen, Q.P., Lim, K.W., Divakaran, D.M., Low, K.H., Chan, M.C.: GEE: A Gradient-based Explainable Variational Autoencoder for Network Anomaly Detection. 2019 IEEE Conference on Communications and Network Security (CNS) (2019).