The anomaly detection and regime searching from the fitness-tracker data

Art Prosvetov[1,2]

[1]LANIT, Moscow, Russia
[2]Space Research Institute Russian Academy of Science, Russia

Abstract. In the current work we describe the developed approach for the problem of human activity monitoring based on data from sensors attached to the hands of various workers. The Gaussian Process model was applied to fill in the gaps in time series and extract outliers. The comparison of several models for activity recognition was performed. An anomaly detection approach was applied that provided useful results for activity monitoring during construction work. In addition, the neural network based on the architecture of Variational Autoencoder allowed us to estimate the main work regimes. The fitness tracker time series dataset was collected, tagged and published for further research.

1 Introduction

In our project, we solve the problem of human activity monitoring based on data from sensors attached to the hands of various workers. First of all, the recognition results help to increase labor productivity and optimize production processes at a building site. Also, the analysis of the behavior of workers allows us to track a person's well-being, compliance with safety measures and accident prevention.

During time series processing the problem of regime extraction regularly appears. By the regime we understand the stable state, which is characterized by their own probability distributions, and the transition of one regime to another is governed by another process or variable. While supervised methods of machine learning can help in time series classification or regression, one has to use unsupervised methods in order to extract regimes in multicomponent time series data.

One of the approaches to search the regimes can be based on the Change Point Detection methods like Pruned Exact Linear Time [1], Optimal detection method [2], or Binary segmentation method [3]. However, Optimal detection methods require the number of change points, and some of CPD methods are hard to use on large multi component data sets with a high number of components due to it’s computationally complexity.

One can use the statistical properties of components, searching the stationary periods in time series and combine them in regimes, however one has to assume that all components should pass one of stationary tests, while in regimes from real world data only part of time series components meet mentioned properties and only domain experts can help in the process of component selection.

Nowadays neural networks successively solve many problems in time series analysis and we suggest to use them in regime extraction.

2 Data Processing

The builders on site use wrist sensors to record the movement of workers' hands. Once a day, the accumulated data from the sensors are uploaded to a server located at the construction site. At the stage of data collection, in addition to sensors, video cameras are also used; during the maintenance only sensors are used. The received data is used by our assessors for markup. Based on obtained labels we train the models for recognizing human activity. After the model testing we implement it in production and regularly recognize the daily data flow. Despite the simplicity of the scheme, we had to face many pitfalls and unexpected problems.

The proof of concept was prepared on data from the test stand. The test stand data was collected during 3 month from a large stream of volunteers ready to hammer in a nail, drill a hole with a drill, or tighten a couple of nuts. A volunteer performed mentioned actions with a fitness-tracker on the hand at a workbench.

First of all, the data of the accelerometer and gyroscope was used. Additionally, we use GPS, barometer and heart rate data.

The accelerometer and gyroscope allowed us to obtain raw data in three coordinate axes with a frequency of 50 Hz, corresponding to a period of 0.02 s. Thus, for recognition, we have six time series, however, for technical reasons, the obtained series turn out to be with gaps and a high level of noise. If we plot a graph showing the gaps between successive measurements, we get the following picture (fig. 1):

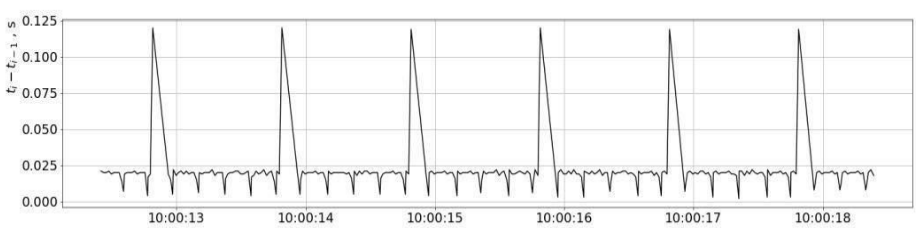

Fig. 1. The gaps between measurements: the time difference between two sequent measurements as a function of time. If the time bins have equal periods, the difference should have a constant form.

The problem of filling in gaps and suppressing noise often arises in time series analysis and has many solutions. Models of the Gaussian process [4] allowed us to solve the problem with gaps and noise, while preserving information as much as possible. The

Gaussian process approach has proven itself well, including in work with time series in astrophysics ([5,6]).

After the model of the Gaussian process was built, it became possible to get rid of noise: if the points did not fall into the confidence interval, then they were replaced by the corresponding points from the Gaussian process (fig. 2).

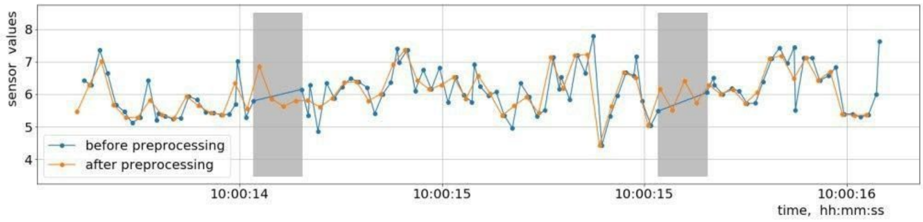

Fig. 2. Time series preprocessing using a model of the Gaussian process. The gaps (grey area) without measurements are filled in by the model values. The outliers are replaced by the corresponding points from the Gaussian process.

The quality of recognition of actions using a neural network on data with preprocessing and without data processing will differ. In our case, the weighted f1-measure grows up from 0.88 to 0.92 (Table 1).

Table 1. The comparison of classification metrics of the models based on neural networks before and after the processing using the model of Gaussian Process.

Action Type	**Before Processing, F1-score**	**After Processing, F1-score**
Hummer using	0.88	0.91
Draw up a bolt	0.91	0.92
Screwdriver using	0.90	0.94
Relaxation	0.83	0.90
Total	**Accuracy: 0.88**	**Accuracy: 0.92**

Assessors are engaged in the data markup using the video of the workflow, thus we have the labeled data and reduce the recognition of activity to the time series classification.

The preparation of the training set is as follows: we divide the multicomponent time series into intervals of the same length, at each interval we select the class label according to the maximum sum of the lengths of the marking intervals that fall into the interval.

As expected, there is a human factor in the data, for example, putting on the watch "upside down". It turns out to be easy to deal with this factor: the classification model determines with an accuracy of more than 90% whether the watch is worn correctly by the worker. In the case of improperly worn bracelets, linear transformation of the raw data enables the same models of activity recognition to be used.

3 Results

3.1 Activity Recognition: the comparison of supervised methods.

In experiments on the test stand, we compared classical algorithms based on automatically generated features and neural networks. Surprisingly, neural networks were unable to significantly bypass the gradient boosting [7] in our case, which may be due to noise in the data and a very limited size of training set. For neural networks, we've tried refined time series, difference schemes, spectrograms, 1D and 2D convolutional layers [9], recurrent layers [10], and combinations of recurrent and convolutional layers. However, the best result is achieved using the gradient boosting classification (lightGBM package [11]) (Table 2). However the neural networks prove to be useful in passing tasks, such as lunch break segmentation or regime estimation.

Table 2. The comparison of classification metrics of the models based on various algorithms.

Model	Macro avg. F1-score	Weighted avg. F1-score
Logistic Regression	0.72	0.74
Neural network	0.92	0.92
Random Forest	0.93	0.93
Gradient Boosting	0.93	0.94

3.2 Hierarchy of actions

As a result of a series of experiments at the construction site, we came to the conclusion that we divide the actions into two levels.

The lower level, consisting of elementary actions. Example: hitting with a hammer, moving with a wrench. The typical scale for the intervals of the lower level is about 5 seconds.

The upper level, consisting of the employee's actions in terms of the goal of the activity. Example: preparation for work, plasterer work, welding, etc. The typical time scale for the upper level intervals is about 30-60 seconds.

The result is a picture of the employee's successive actions throughout the entire working day, with details down to elementary movements.

3.3 Anomaly detection

Various and often unpredictable situations can occur on a construction site. During the implementation of the project on the construction site, we should avoid to use classification models on actions that are obviously outside the scope of the observed behavior on the training set. In order to be ready to face actions outside of the used classes, we used the anomaly detection.

The anomaly detection helped us find:

- errors of assessors;
- atypical behavior of workers;
- the emergence of new elements in the technical process;

- identification of "suspicious" employees.

If one uses the Isolation Forest [12] algorithm on the same features that are calculated for the main classification model, it is possible to obtain a “anomaly score” for each object based on mean anomaly score of the trees in the forest: a numerical value that characterizes the degree of typicality for each object in the sample. The measure of normality of an observation given a tree is the depth of the leaf containing this observation, which is equivalent to the number of splittings required to isolate this point. The higher the score, the more typical object in the sample is its owner. The distribution of the anomaly score is shown on fig. 3.

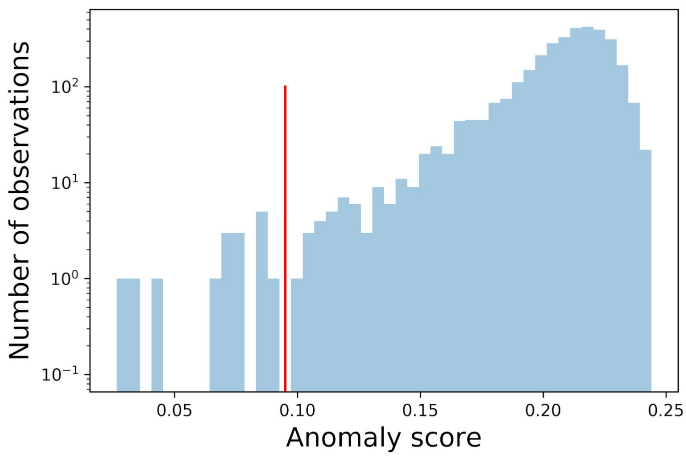

Fig. 3. The distribution of anomaly score obtained by Isolation Forest. Red line is a threshold for anomaly decisions, obtained on a validation dataset using the Isolation Forest model, fitted on the train dataset.

For the next step, it is important to choose a threshold value, starting from which it will be possible to determine whether an object is an anomaly by the normality rating. In this question, one can use estimation from the expected frequency of occurrence of anomalies, or choose a threshold value for some additional considerations. We have chosen a threshold value based on the distribution of the anomaly score: the figure 3 shows that, starting from a certain value, the nature of the distribution of anomaly score in validation dataset changes. The obtained threshold isn't based on the expected proportion of outliers in the dataset and is allowed to obtain flexible values based on the real distribution of objects.

Table 3. The comparison of two types of workers. Typically the worker produces 0-5 number of anomalies in 30-seconds intervals during the day.

Worker Type	Number of anomalies during the day
Typical	0-5
Abnormal	> 10

An important point is the following observation: the anomaly detection can be efficiently applied for each class of activity separately, otherwise rare classes of actions are distinguished as an anomaly. We trained an independent model for each class of activity: work, moving, relaxing and preparing. Typically each worker produced several anomalies in each class of actions. Summing up the anomalies in all classes we were able to find the worker with the highest number of anomalies. After the inspection we made a conclusion that the source of anomalies for the plasterer worker was based on the sex of worker: the plasterer with highest number of anomalies was male, while only female plasterers were present in the training set. The comparison of typical and anomaly workers can be found in Table 2.

Using the number of anomalies per hour as a criterion, it is possible to identify intervals at which the employee evades the work. One of the 1 hour intervals with a high number of anomalies was associated with the movement of a drunk worker holding onto walls. However, the work activity of the drunken worker didn't present a significant number of anomalies. Another interval with a high number of anomalies in movement was associated with a case in which a plasterer shrinked away from the work during 2-3 hours.

While the majority of worker's actions can be recognised using classification models, we still need to resolve cases with duty evasion. For example, let's consider two cases: first one, when the plasterer evades the duty and the second one, when the plasterer makes a lot of transporting work. Both periods will be recognised by the model as periods with high proportion of movement and relaxation, but only in case of the work evader the proportion of anomalies in movement per hour will be the highest.

3.4 Regime Searching

In order to obtain generalization of time series we used a neural network with architecture of variational autoencoder [13]. The input layer was applied on the tensor of parameter evolution during one time frame: the period of the time frame was 30 seconds.

Encoder had 4 convolution layers (2d) with 32 filters on the first layer and 64 filters on the 3 sequent layers. Kernel size was equal to 3, and "ReLu" activation function. A fully connected layer was applied after convolution layers with 32 neurons and ReLu activation function.

The latent dimension layer had a size equal to 2, that allowed us to visualize the results.

Decoder had an input layer of 2 neurons, a fully connected layer with ReLu activation function. The next decoder layer was a fully connected layer with the size equal to M*N/2, where M and N are the shape of input tensor. Next decoder layer was the Transposed convolution layer also known as Deconvolution with 16 filters and kernel size equals 3. The need for transposed convolutions arises from the desire to use a transformation going in the opposite direction of a normal convolution. Finally the decoder had a Convolution layer to obtain the tensor size equal to the input tensor size.

For the loss function we used a reconstruction term and a regularisation term in form of Kullback-Leibler divergence. The reconstruction term in loss function was based on binary cross entropy loss. The regularization term in the form of Kullback-Leibler divergence was used. Also we used a reparametrisation trick [14] to make the backpropagation possible through the network.

We trained neural networks during 2-3 epoches, while loss decayed on validation dataset. After model training we were able to use the encoder to transform multidimensional time series into a 2-dimensional latent space. The 2D projection of time series were grouped by clusterization algorithms (Agglomerative Clustering [15] or Gaussian Mixture model) that allowed us to obtain regimes. The Gaussian Mixture model implements the expectation-maximization algorithm [16] for fitting the data with mixture-of-Gaussian models. The labels for worker actions in high level actions were obtained by assessor's markup. The obtained label allowed us to understand if the cluster has common properties and group clusters with the same dominant label. The resulting distribution of points in latent space by the clusters can be found on fig. 4.

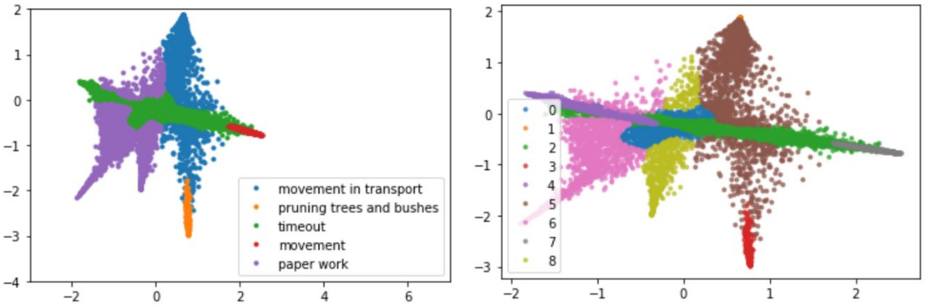

Fig. 4. The 2D projection on latent space of time series data from the fitness-tracker. Right diagram: the clusterization of time intervals in latent space of variational autoencoder. Left diagram: the union of clusters based on the dominant labels of short-time activity, obtained from assessor's markup. The dominant labels of short-time activity in the resulting cluster are presented in legend.

The obtained regimes allowed us to estimate the main classes of work on high level hierarchy on the early stage of data collection.

4 Conclusions

In the current work we are trying to accumulate the experience of human activity recognition using fitness-tracker data. During our experiments we found the following:

- Neural networks didn't present the significant metric uplift in supervised regime on the data from the accelerometer and gyroscope in comparison with gradient boosting in the task of activity recognition. We suggest that neural networks require a significantly higher number of observations in the dataset to demonstrate uplift.
- The usage of Gaussian Process model in data preprocessing allows to increase the metrics of the classification model, based on neural networks.

- Anomaly detection algorithms can recognize the difference between human activity and allow us to identify the work evaders. In one case the drunken behavior was found using an anomaly detection model.
- The regime estimation can be based on results of variational autoencoders, trained on the multicomponent time series frames.
- The researchers interested in the fitness tracker dataset collected on volunteers can access the data using the following link: (will_be_added_after_review).

References

1. R. Killick, P. Fearnhead, and I. A. Eckley, Journal of the American Statistical Association, 107(500):1590–1598 (2012).
2. S. M. Kay and A. V. Oppenheim. Fundamentals of Statistical Signal Processing, Volume II: Detection Theory. Prentice Hall (1993).
3. A. Sen and M. S. Srivastava. The Annals of Statistics, 3(1):98–108 (1975).
4. C. E. Rasmussen and C. K.I. Williams, MIT Press (2006)
5. D. R. Wilkins, MNRAS 489, 2, 1957–1972 (2019)
6. M. V. Pruzhinskaya, K. L. Malanchev, M. V. Kornilov, E. E. O. Ishida, F. Mondon, A. Volnova and V. S. Korolev, MNRAS 489, 3, 3591–3608 (2019)
7. M. Christ, N. Braun, J. Neuffer and A.W. Kempa-Liehr (2018). Neurocomputing 307 72-77 (2018)
8. Friedman, J. H. "*Greedy Function Approximation: A Gradient Boosting Machine*" (1999)
9. Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard and L. D. Jackel, Neural Computation, 1(4):541-551, Winter (1989).
10. Williams, Ronald J.; Hinton, Geoffrey E.; Rumelhart, David E. Nature. 323 (6088): 533–536 (October 1986).
11. Guolin Ke, Qi Meng, T. Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, Tie-Yan Liu. Advances in Neural Information Processing Systems 30 (NIPS 2017), pp. 3149-3157.
12. Liu, Fei Tony, Ting, Kai Ming and Zhou, Zhi-Hua. ACM Transactions on Knowledge Discovery from Data (TKDD) 6.1 (2012): 3.
13. An, Jinwon and S. Cho. Special Lecture on IE (2015).
14. Diederik P Kingma, Max Welling arXiv:1312.6114 [stat.ML] (2013).
15. Rokach, Lior, and Oded Maimon. Data mining and knowledge discovery handbook. Springer US, 321-352 (2005).
16. Dempster, A.P.; Laird, N.M.; Rubin, D.B. Journal of the Royal Statistical Society, Series B. 39 (1): 1–38 (1977).