Determining mouse behavior based on brain neuron activity data

Anastasia Vodeneeva[1], Iosif Meyerov[1], Yury Rodimkov[1], Mikhail Ivanchenko[1], Vladimir Sotskov[2,3], Mikhail Krivonosov[1], Konstantin Anokhin[4]

[1] Lobachevsky State University of Nizhny Novgorod, Nizhny Novgorod, Russia
[2] Center of Interdisciplinary Research in Biology, Collège de France, Paris, France
[3] Institute of Biology, École Normale Supérieure, Paris, France
[4] Institute for Advanced Brain Studies, Lomonosov Moscow State University, Moscow, Russia

Abstract. The study of the relationship between brain neuron activity and behavioral responses of humans and other animals is an area of interest, although it has received relatively little attention from scientific biology and medical research centers. In this paper, we consider the problem of determining a mouse position in a circular track based on its neural activity data, and investigate the use of machine learning for solving this problem. The study is conducted in two parts: a classification task, where the model predicts which sector of the track the mouse is in at a particular time, and a regression task, where it predicts exact coordinates for each time step. We propose a neural network-based solution for both tasks, based on a graph of brain neuron activity. Accuracy results were obtained: 89% for classification and 93% for regression.

Keywords: brain, neural activity, artificial intelligence.

1. Introduction

The mechanisms underlying brain function and human and animal behavior comprise one of the most significant areas of research within modern science. Complexity, variability, and motivation are the most vital characteristics of the behavioral patterns of living organisms [1]. In this study, we explore the possibility of predicting an organism behavior based on neural impulses using machine learning (ML) tools. The experiment involves a mouse placed on a circular track and freely moving within it. Brain neuron impulses were recorded using a head-mounted NVista HD miniscope [2], which could detect calcium signals from neurons. Cell images were captured using a set of genetically engineered calcium indicators [3,4]. The mouse with the miniscope was placed on a track that had been previously cleaned of foreign odors. There are four marks along the mutually perpendicular diameters of the track, which allow the animal to draw any conclusions about its current position. At each point in time during the experiment, the coordinates of the mouse position are recorded. The video recording frequency is 20 frames per second, and its total duration is 15 minutes and 39 seconds. Data for the experiment was obtained in article [5].

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63772-8_11

The main scientific interest in this problem is the ability to determine the coordinates of a mouse location based on impulses from brain neurons using a graph of neural connections and ML methods. The potential of artificial intelligence to analyze and replicate the intelligence of living biological beings offers many opportunities for biological and medical research. This ability forms the basis for our work. In the course of our research, we answer the following questions:

1. Is it possible to construct such an artificial neural network architecture that allows tracking the mouse coordinates with acceptable accuracy based on calcium activity in mouse hippocampal neurons?
2. How well can models be trained using existing data of movement trajectories and neural activity during movements?
3. Which of the two mathematical formulations of the ML task is more suitable for solving this problem?

The article is organized as follows: Section 2 reviews background and related work; Section 3 and Section 4 describes the process of solving this problem through classification and regression, respectively. Section 5 concludes the article.

2. Background and Related Work

Despite widespread interest in this problem in the fields of biology and medicine, very little research has been published on this subject.

In [6], authors described an ML method for analyzing the behavior of mice kept in groups up to four individuals for several days in a controlled environment in real time. It was described how this method can be used to study the effects of mutations in genes linked to autism on mouse behavior. In [7], ML techniques were used to distinguish between different mouse conditions based on brain activity and camera data. The aim of the study is to develop a learning approach that could accurately reflect classification results and transfer those results to other mouse conditions.

The work [8] demonstrated that continuous behavioral data can be analyzed using approaches similar to natural language processing. This data supports further research into detecting complex pathophysiological alterations accompanied by changes in the behavioral profile.

The work [9] explored the solution to determining the movement of a mouse based on data from brain neurons activity using a statistical approach without prior knowledge. The authors hypothesized that, when combined with innovative techniques for estimating coordinates, a created Bayesian model could extract data about complex behavior [10,11]. In [12], the authors solved this problem by reconstructing time series of brain cell activity and identifying fields that constitute cognitive maps. The data was used in the form of a three-dimensional graph of cellular connections, based on an algorithm for reconstructing the dynamic graph of calcium event distribution, with two dimensions being the number of cells in the studied part of the brain and the third being the number of studied time points [13]. The reconstruction of these graphs was done using calcium events from neurons

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63772-8_11

detected using the algorithm described in [5], which was also used in our work to obtain the data.

3. Classification of mouse position on a circular track

In our study, we classify the mouse position on a circular track by dividing it into sectors. We solve this problem by determining whether a given mouse position belongs to a particular class (sector) of the track. The object of this task is the coordinates of the mouse position angle at different points in time, and the class is a specific sector of the track that the mouse travels along. The set of vertices in the graph is the number of brain cells, which is 562, and the set of edges represents the connections between these cells. The total number of graphs in our dataset is 18 775, corresponding to the number of measurements taken at different times. The response is the angle α of the mouse at each time point (see Fig. 3).

We use a convolutional neural network (CNN) to solve the classification problem of determining the sector of the mouse position on a circular track. The sum of the squared differences between the output signals from the network and their required values is used as a measure of how well the network performs (MSE, mean squared error):

$$R_{MSE} = \frac{1}{n}\sum_{i=1}^{n}\left(\alpha_i - \alpha_i^{predict}\right)^2, \tag{1}$$

where n is a number of classes, α_i is a real angle of the mouse position, and $\alpha_i^{predict}$ is a predicted angle. CNN is used to solve classification problem, based on an example from article [14]. CNN has a structure shown in Fig. 1a. The first 75% of data in time is taken for training, and the remaining 25% is taken for testing. For some points in time, a visualization of the neuron connections is created to obtain a clearer picture of what is happening (see Fig. 2). To generalize results, we use a function that calculates the error as the ratio of difference in real and predicted values to circumference (RGE, resulting generalized error):

$$R_{RGE} = \frac{|\alpha - \alpha^{predict}|}{2\pi} \times 100\,\%\,, \tag{2}$$

where r is a radius of the track, α is a real angle of the mouse position, $\alpha^{predict}$ is a predicted angle (all angles are taken in radians).

To compare results obtained in two different cases, classification results are converted to regression results by finding median value for each class:

$$R_i = \frac{\min\left\{\left|Y_i - Y_i^{predict}\right|, n - \left|Y_i - Y_i^{predict}\right|\right\}}{n}, \tag{3}$$

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63772-8_11

where n is a number of classes, Y_i is a real number of the sector, $Y_i^{predict}$ is a predicted number. Here and below, all errors are given for the test set.

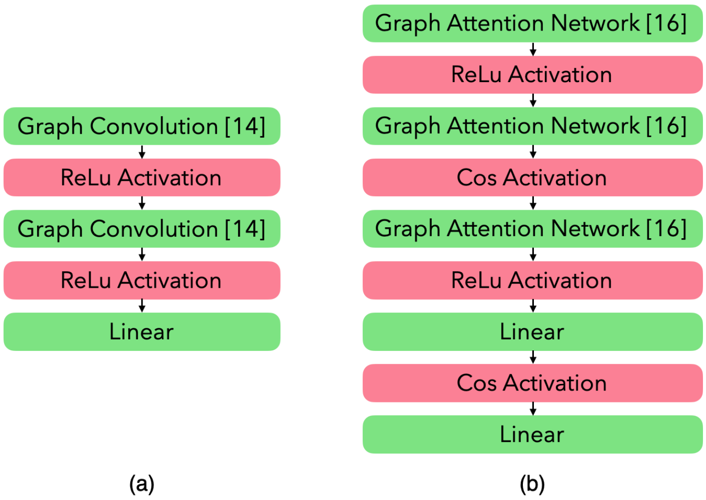

Fig. 1. Layer-by-layer structure of CNN *(a)* and GNN *(b)*.

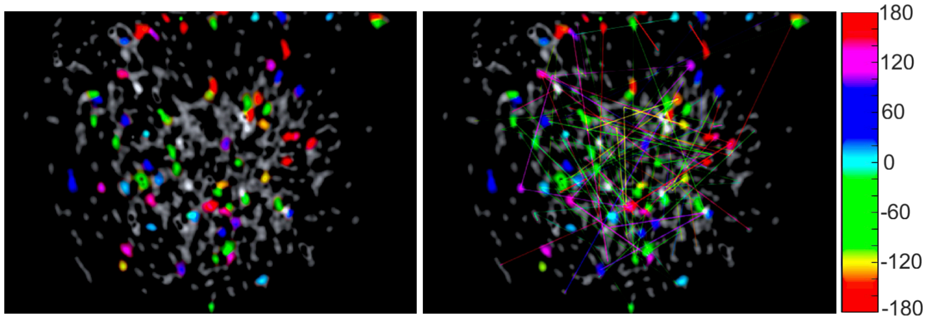

Fig. 2. Images of brain graphs at two moments of time (5 sec and 25 sec from the start). Cells with the same color are activated at the position with that color on the scale (in degrees).

We started by trying to locate the mouse by dividing the circle into two parts and identifying the halves. Here, we could achieve an RGE of 18%. Next, we attempted to predict the quarter in which the mouse is located, with an RGE of about 19%. For dividing into eight parts, the RGE was 22%; for dividing into twelve parts, it was 25%. Since the mouse size is approximately 8.3% of the circumference, solving the classification problem makes sense if the number of classes does not exceed 12 (see Fig. 3). Additionally, removing intervals where the mouse moves less than 8.3% of the way around the circle reduced the RGE to 14%. It was suggested that if all intervals with constant positions are removed from the dataset and CNN is trained on this new dataset, a smaller error could be achieved. We also considered that the

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63772-8_11

network architecture was too simplistic for training on such a complex task. Therefore, we decided to increase the number of hidden layers and modify the activation functions accordingly. Additionally, the experimental results indicated that the main challenges in the model performance stemmed from the class boundaries. To address this, we changed the problem formulation from classification to regression, as we hypothesized that this would reduce the overall error rate of the solution by eliminating class boundaries themselves. In order to test whether this approach could improve the results achieved for classification, we formulated a regression task.

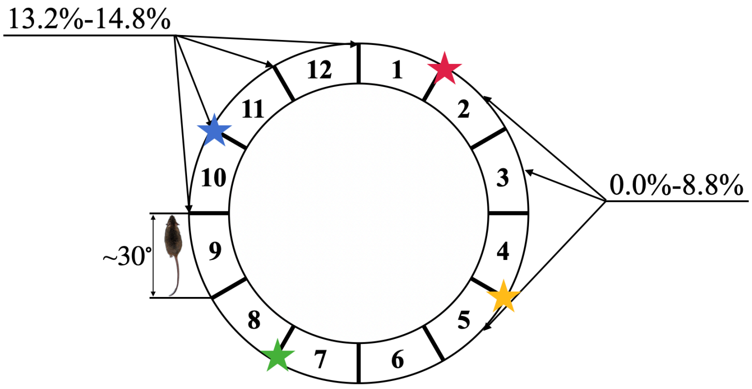

Fig. 3. The case of dividing a track into 12 sectors. The sector measure is 30°, and the length of a mouse is 30°. Colored dots indicate the physical markers for self-identification of mouse.

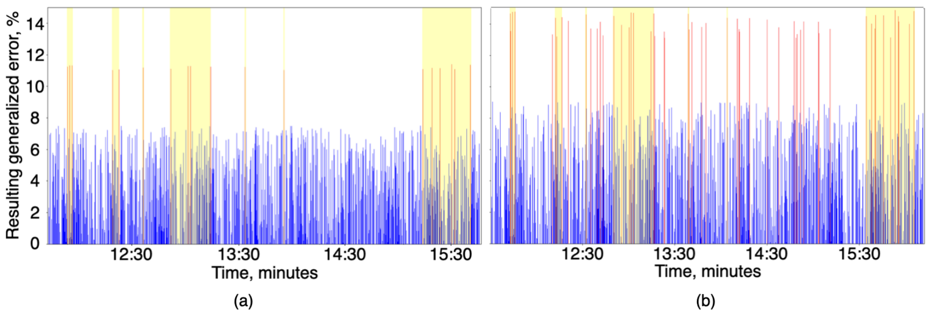

Fig. 4. Error plot for regression *(a)* and classification *(b)*. The vertical axis represents RGE in percentage terms, the horizontal axis represents time from the start in minutes. The largest errors are shown in red; the intervals where the mouse moves less than 8.3% of the circumference are shown in yellow.

4. Regression of mouse position on a circular track

For our study, regression involves determining the exact angle of the mouse position at each point in time. The set of brain neuron impulses at each time point is considered to be a set of features. The change in the mouse location on the track is the de-

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63772-8_11

pendent variable. The angle of the mouse coordinate at each moment in time is the output. We chose the mean absolute error (MAE) as an error metric:

$$R_{MAE} = \frac{1}{n}\sum_{i=1}^{n} |\alpha_i - \alpha_i^{predict}|, \tag{4}$$

where n is a number of predicted values, α_i is a real mouse position angle, $\alpha_i^{predict}$ is a predicted angle [14]. The structure of a graph neural network (GNN) used to solve a regression problem is shown in Fig. 1b. MSE (1) and MAE (4) are used as loss functions. The error obtained using MAE is smaller than the error using MSE. After that, GNN is used for classification and produces better results than CNN.

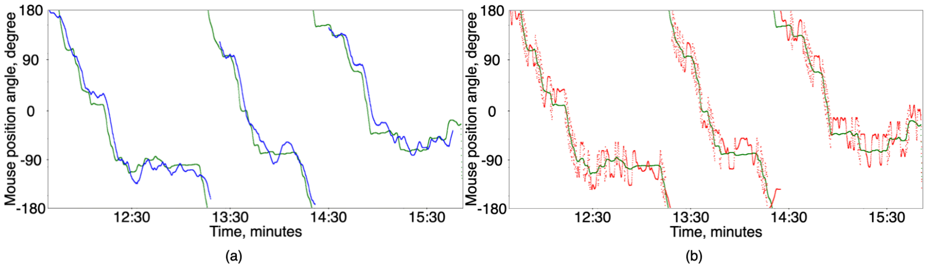

Fig. 5. Average coordinate in a sliding window of 5 seconds (100 frames wide) *(a)* and the coordinates at each time point *(b)*. The vertical axis represents mouse position angle, horizontal axis represents time from the beginning in minutes. Actual coordinates are green; predicted coordinates are blue on *a*, are red on *b*.

The plot of error changes on test data is shown in Fig. 4a. As can be seen from this plot, the maximum regression RGE does not exceed 12%, which is a better result than for classification. In addition, by looking at areas of constant mouse position, we concluded that error peaks occur exactly at moments when the mouse stops or starts moving. During all other time intervals, RGE does not exceed 7%. Based on results from solving regression task, plot of predicted mouse coordinate was drawn in Fig. 5b. As expected, because of absence of dividing the track into sectors errors at boundaries disappeared, but because of unpredictable behavior during constant coordinates intervals, maximum RGE equals to 7%. For comparative analysis, dynamic of RGE changes for classification task was also plotted. Plot of classification error changes is shown in Fig. 4b. Here we show that the maximum RGE for classification is greater than for regression, being approximately 15% (which is better than previously obtained). Additionally, it is clear that error reaches this maximum not only at constant positions, but also between them. This occurs at the boundaries between classes when mouse moves from one class to another. Therefore, for classification, the maximum RGE is 11% and for regression 7%. By changing the problem formulation from classification to regression, accuracy increases by 4%. A plot shows the mouse movement curve based on original data and predicted data in regression (see Fig. 5b). The predicted coordinate generally follows the real

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63772-8_11

coordinate dynamics, but has larger fluctuations at intervals with small changes in real coordinates. For more detailed conclusions, a trajectory of the average coordinate was plotted in a sliding window of 100 frames (5 seconds). This plot is presented in Fig. 5a. The moving average is calculated using an interval $[t - w, t]$, where w is the window size, and t is the averaged data argument. The maximum RGE here is 8%. We conclude that in a window of this width, the predicted trajectory closely follows the actual one.

Based on all the results obtained, Table 1 was compiled showing the values of maximum RGE for two ML problem formulations on two network types (see Fig. 1) using the train and test datasets.

Table 1. Generalizing table of the RGE errors for two formulations of problems when solving them using two methods for constructing a neural network with train and test data.

	Classification		Regression	
	CNN (Fig. 1a)	GNN (Fig. 1bt)	CNN (Fig. 1a)	GNN (Fig. 1b)
Train	13 %	8 %	9 %	4 %
Test	22 %	15 %	13 %	7 %

5. Conclusions

As a result of the experiments, GNN was found to solve a classification task with an RGE (2) of 11% and a regression task with an RGE (2) of 7%, respectively. Thus, it would be advisable to formulate and solve this problem in terms of regression analysis. Given that the problem addressed in this work has not been widely studied, it is impossible to determine with certainty the minimum error that could be achieved with this data. Additionally, it remains uncertain whether we have all the necessary information available to construct a high-quality neural network. It is now essential to interpret these findings from a neurobiological standpoint and develop a strategy for improvement. This approach should be based on real biological processes, and its results should be applied to a larger number of mice in order to test whether the patterns observed in one mouse apply to others. Does the network trained on the first day of the experiment produce the desired level of accuracy when re-run on the second and third days? What patterns exist between the graphs of neuron activity on different days? We plan to investigate these questions as part of our future research.

Acknowledgments. This research was funded by the "Center of Photonics" funded by the Ministry of Science and Higher Education of the Russian Federation (contract no. 075-15-2022-293).

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63772-8_11

Data and Code Availability. All data, code and launch scripts used for the article is provided as part of the replication package. It is available at https://github.com/nastyalabs/mouseBrain.

References

1. Andrianov, V. Neurons, brain and behavior. Bulletin of the International Academy of Sciences. Russian section 2 (2007): 25-29.
2. Ghosh, K.; Burns, L.; Cocker, E.; Nimmerjahn, A.; Ziv, Y.; Gamal, A.; Schnitzer, M. Miniaturized integration of a fluorescence microscope. Nat. Methods 2011, 8, 871–878.
3. Barykina, N.; Sotskov, V.; Gruzdeva, A.; Wu, Y.; Portugues, R.; Subach, O.; Chefanova, E.; Plusnin, V.; Ivashkina, O.; Anokhin, K.; et al. FGCaMP7, an Improved Version of Fungi-Based Ratiometric Calcium Indicator for In Vivo Visualization of Neuronal Activity. Int. J. Mol. Sci. 2020, 21, 3012.
4. Subach, O.; Sotskov, V.; Plusnin, V.; Gruzdeva, A.; Barykina, N.; Ivashkina, O.; Anokhin, K.; Nikolaeva, A.; Korzhenevskiy, D.; Vlaskina, A.; et al. Novel Genetically Encoded Bright Positive Calcium Indicator NCaMP7 Based on the mNeonGreen Fluorescent Protein. Int. J. Mol. Sci. 2020, 21, 1644.
5. Sotskov, V., et al. The Rapid Formation of CA1 Hippocampal Cognitive Map in Mice Exploring a Novel Environment. Advances in Cognitive Research, Artificial Intelligence and Neuroinformatics: Proceedings of the 9th International Conference on Cognitive Sciences, Intercognsci-2020, October 10-16, 2020, Moscow, Russia 9. Springer International Publishing, 2021.
6. De Chaumont, F., et al. Real-time analysis of the behaviour of groups of mice via a depth-sensing camera and machine learning. Nature biomedical engineering 3.11 (2019): 930-942.
7. Mantripragada, Sh., Dionne, E., and Chen, J. Identifying Behavioral Movements of a Mouse Using Machine Learning. Columbia Junior Science Journal Spring 2020 (2020): 3.
8. Gharagozloo, M., et al. Machine learning in modeling of mouse behavior. Frontiers in neuroscience 15 (2021): 700253.
9. Weissenberger, Y., King, A. J., and Dahmen, J. C. Decoding mouse behavior to explain single-trial decisions and their relationship with neural activity. bioRxiv (2019): 567479.
10. Kingma, D. P. and Welling, M. Auto-Encoding Variational Bayes. Iclr 1–14 (2014).
11. Rezende, D. J., Mohamed, S. and Wierstra, D. Stochastic Backpropagation and 316 Approximate Inference in Deep Generative Models (2014).
12. Sotskov, V., et al. Calcium imaging reveals fast tuning dynamics of hippocampal place cells and CA1 population activity during free exploration task in mice. International journal of molecular sciences 23.2 (2022): 638.
13. Mitroshina, E., et al. Novel Algorithm of Network Calcium Dynamics Analysis for Studying the Role of Astrocytes in Neuronal Activity in Alzheimer's Disease Models. International Journal of Molecular Sciences 23.24 (2022): 15928.
14. Defferrard, M., Bresson, X., and Vandergheynst, P. Convolutional neural networks on graphs with fast localized spectral filtering. Advances in neural information processing systems 29 (2016).
15. Nunez, E., Steyerberg, E. W., and Nunez, J. Regression modeling strategies. Revista Española de Cardiología (English Edition) 64.6 (2011): 501-507.
16. PaddlePaddle, https://github.com/PaddlePaddle/PGL, last accessed 2024/02/17.

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63772-8_11