*Article*

# Transformer-based Methods for Neural Decoding

**Haonan He** *

School of Automation Science and Engineering, South China University of Technology, Guangzhou 510641, China; auhaonanhe@mail.scut.edu.cn

* Correspondence: auhaonanhe@mail.scut.edu.cn; Tel.: +86 19927527314

**Abstract:** Neural decoding from spiking activity is an essential tool for understanding the information encoded in population neurons, especially in applications like brain-computer interface (BCI). Various quantitative methods have been proposed and have shown superiorities under different scenarios respectively. From the machine learning perspective, the decoding task is to map the high-dimensional spatial & temporal neuronal activity to the low-dimensional physical quantities (e.g., velocity, position). Because of the complex interactions and the abundant dynamics among neural circuits, good decoding algorithms usually have the capability of capturing flexible spatiotemporal structures embedded in the input feature space. Recently, the Transformer-based models are widely used in processing natural languages and images due to its superior performances in handling long-range and global dependencies. Hence, in this work we examine the potential applications of Transformers in neural decoding and introduce two Transformer-based models. Besides adapting the Transformer to neuronal data, we also propose a data augmentation method for overcoming the data shortage issue. We test our models on three experimental datasets and their performances are comparable to the previous state-of-the-art (SOTA) RNN-based methods. In addition, Transformer-based models show increased decoding performances when the input sequences are longer, while LSTM-based models deteriorate quickly. Our research suggests that Transformer-based models are important additions to the existing neural decoding solutions, especially for large datasets with long temporal dependencies.

**Keywords:** Transformer; spike; neural decoding; CNN; RNN; LSTM; deep learning; information; neuroscience

## 1. Introduction

Neural decoding studies the relationship between neural population activities and the outside world. It is a central tool to understand how neurons encode external variables and can facilitate engineering applications such as brain-computer interfaces (BCI). In essence, neural decoding is to find a mapping relationship between neuronal data and particular variables (velocity, position, etc.) observable in the outside world. Such relationship was traditionally described by linear methods in the past. Recently machine learning methods especially those based on neural networks have been widely used [1], facilitating neural decoding from various aspects.

Because of the complex interactions and the abundant dynamics among neural circuits, good decoding algorithms usually have the capability of capturing spatiotemporally dependent structures embedded in the input feature space. Recurrent Neural Networks (RNNs) are by far the most common deep learning architectures applied in neural decoding due to its ability of dealing with sequentially dependent data [1]. Among them LSTMs are the most commonly used because they are able to learn long-range dependencies better than other recurrent structures [1–6]. Convolutional Neural Networks (CNNs) are also frequently used for decoding neural signals in the form of fMRI image, calcium image or multi-channel EEG waves [7–9] because they are able to learn local dependencies of the data. Although most of these deep learning algorithms have achieved better performances compared with traditional machine learning methods, they still suffer from problems such as gradient vanishing and the weakness of extracting global features.

Transformer is a new neural network structure that has been widely used in machine learning community in recent years. It has achieved state-of-the-art (SOTA) performances in tasks such as natural language processing [10], object detection [11], image classification [12] and protein engineering [13], etc., suggesting its wide applicability. Transformer uses multi-head attention, enabling it to avoid information loss over time steps compared to recurrent structures and giving it wider receptive fields than convolutional layers. Such attention mechanism suggests Transformer's superior ability of handling long-range and global dependencies. However, Transformer-based models are still relatively little used in neural decoding.

In this work we explore possibilities of applying Transformers in neural decoding and introduce two Transformer-based models, Spatial Temporal Transformer (STT) and Convolutional Spatial Temporal Transformer (CSTT). In experimental datasets of recordings from monkey motor cortex, monkey somatosensory cortex, and rat hippocampus, our Transformer-based models achieve performances comparable to the previous SOTA RNN-based methods. Besides adapting the Transformer to spikes, we also propose a data augmentation method based on Generalized Linear Model (GLM) to generate synthetic neuronal datasets larger than real ones. We test our models on three augmented datasets and they show better decoding performances. In addition, Transformer-based models show stable and increased decoding performances when the input sequences are longer, while LSTM-based models deteriorate quickly. Our research suggests that Transformer-based models might be an alternative in neural decoding, especially for large datasets with long temporal dependencies.

## 2. Materials and Methods

### *2.1. Dataset*

We used the same three datasets as in [1], which were separately collected from motor cortex, somatosensory cortex and hippocampus. In the task for decoding from motor cortex, monkeys moved a manipulandum that controlled a cursor on a screen [14], and we aimed to decode the x and y velocity of the cursor. The data were 21 min and contained 164 neurons. The mean and median firing rates were 6.7 and 3.4 spikes/s respectively. Data were put into 50 ms bins. We used 700 ms of neural activity (the concurrent bin and 13 bins before) to predict the current movement velocity.

Dataset recorded from somatosensory cortex [15] was from the same task. It contained data of 51 min and 52 neurons. They were put into 50 ms time bins. The mean and median firing rates were 9.3 and 6.3 spikes/s respectively. We used 650 ms surrounding the movement (the concurrent bin, 6 bins before, and 6 bins after) to predict the current movement velocity.

Dataset recorded from hippocampus came from the task that rats chased rewards on a platform[16,17], and we aimed to decode the x and y position of the rat. This dataset contains data over a period of 75 min from 46 neurons. They had mean and median firing rates of 1.7 and 0.2 spikes/s respectively. Data were put into 200 ms bins. We used 2 s of surrounding neural activity (the concurrent bin, 4 bins before, and 5 bins after) to predict the current position.

For all three datasets we performed the same treatment as in paper [1].

### *2.2. Model Structure*

A distinctive feature of the spike signal is that it has both temporal and spatial features, with the temporal features arising from the encoding dynamics of variables in the outside world and the spatial features arising from the interactions between neurons. We perform a simple experiment to confirm this idea. We build a Generalized Linear Model (GLM) for each neuron to get the firing rate of each time bin. Then sample it according to a Poisson distribution to generate fake spikes which do not contain many spatial features. The results are shown in Table 1. We find that the fake spike signals have lost some features compared with the real spike signals, considered to be the spatial features mentioned above. This provides a clue for the design of our Transformer-based model.

**Table 1.** The LSTM performs better on real data than on synthetic neural signals generated by GLM, suggesting that spikes contain spatial features.

| **Dataset** | **Somatosensory** | **Motor Cortex** | **Hippocampus** |
|---|---|---|---|
| | **Cortex ( $R^2$ )** | **( $R^2$ )** | **( $R^2$ )** |
| Synthetic | 0.7578 | 0.6889 | 0.4763 |
| Real | **0.8621** | **0.8826** | **0.6088** |

Transformer was first used in Natural Language Processing (NLP) to take a sentence as the input of the network, where each word is embedded as a token. In the task of neural decoding of spike signals, a natural idea is to take time series as the input and embed each time bin as a token which contains the firing rates of all neurons at this moment. The attention structure in Transformer is concerned with the relationship between tokens, which is appropriate for solving NLP tasks where we need to focus on the relationship between each word. However, for spike's neural decoding task this means that only the temporal information is attended and the spatial features are not sufficiently extracted.

To fully extract the spike signals' spatial features, we embed each neuron's time bins as a token, so that attention can focus on the interactions between neurons. We try to extract spatial and temporal features of spike signals separately with Transformer, or do both sequentially, and the results are shown in Table 2. The results show that Transformer performs best for extracting both spatial and temporal features, which is consistent with the above idea.

**Table 2.** Attend spikes from temporal or spatial axis or both. Temporal Transformer extracts more features than spatial Transformer and combining these two can improve the ability of extracting features.

| **Model** | **Somatosensory** | **Motor Cortex** | **Hippocampus** |
|---|---|---|---|
| | **Cortex ( $R^2$ )** | **( $R^2$ )** | **( $R^2$ )** |
| Temporal | 0.7904 | 0.7762 | 0.5004 |
| Spatial | 0.7413 | 0.4601 | 0.0290 |
| Spatial & Temporal | **0.8153** | **0.8208** | **0.5518** |

Based on the above ideas, we modify Transformer structure and introduce a new model called Spatial Temporal Transformer (STT). Our structure improves the multi-head attention in [18] by enabling the attention layer to extract features of the input data from both temporal and spatial dimensions. Compared to using two Transformers sequentially, our approach reduces network parameters and mitigate information loss over layers. The structure of our model is shown in Figure 1, where two attention heads are used to extract spatial and temporal features from the input data, respectively. And each attention head can be split into multiple smaller attention heads to increase the diversity of the extracted features. These heads are all simply concatenated and linearly transformed to maintain the same shape as the input data.

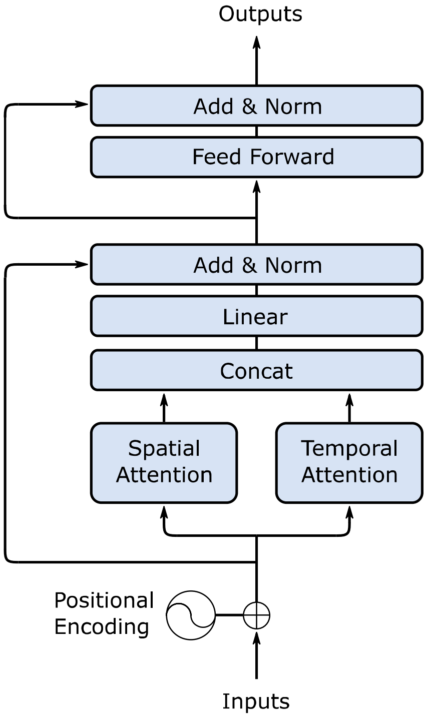

**Figure 1.** Spatial Temporal Transformer (STT) architecture. We do positional encoding only across temporal dimension for its strict chronological order. Both spatial and temporal attention are done to inputs after positional encoding. The results of attention are concatenated and then linearly transformed to keep shape. It is then added with inputs and normalized. In the end there is a position-wise feed-forward network followed by an add & norm module the same as before.

The complete data input process is as follows: the input data is shaped as $S \times T \times N$, where S represents the number of input samples, T represents the length of the input sequence, and N represents the number of neurons. After positional encoding, it is input into the Transformer structure, where the data goes through spatial attention and temporal attention to get spatial head and temporal head respectively. The two heads are concatenated and linearly transformed to keep the original shape $S \times T \times N$. It is then summed with the original input. And a layer normalization is performed to get the output of the attention part, which is put into the position-wise feed-forward part consists of two fully connected layers with a ReLU activation in between. The outputs of attention part and feed-forward part are then summed and layer-normalized to get the final output of the transformer block, keeping the original input of $S \times T \times N$.

To further enhance the feature extraction capability of our model, we introduce a new structure denoted as Convolutional Spatial Temporal Transformer (CSTT) with a convolutional structure before the input of the network, containing two one-dimensional convolution layers to extract features across neurons and time. One is a spatial convolutional layer of size $N \times 1$ and the other is a temporal convolutional layer of size $1 \times 3$. In the temporal convolutional layer, we set padding mode to 'same' to keep number of time bins unchanged. We can change the number of convolutional kernels C to adjust the output shape of convolutional structure. The data is then fed into STT after normalization, activation and dropout layers. The architecture of CSTT is shown in Figure 2.

doi:10.20944/preprints202108.0011.v1

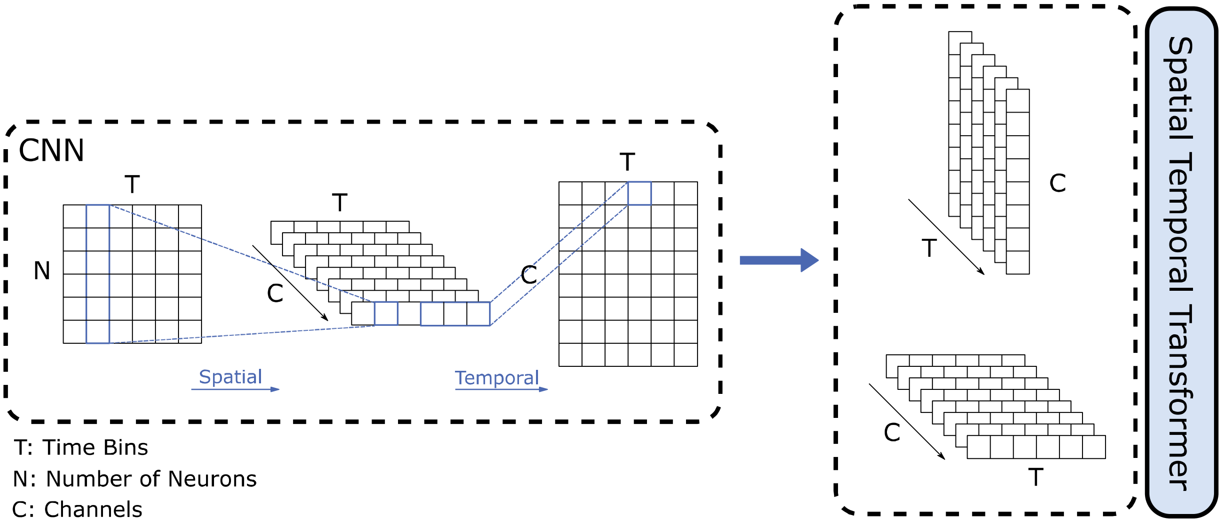

**Figure 2.** CSTT architecture and data shape transformation details. We add a convolutional structure before STT to further enhance its modeling ability. The convolutional part consists of two 1-D convolutional layers, one is a spatial convolutional layer of size $N \times 1$ and the other is a temporal convolutional layer of size $1 \times 3$ . In the temporal convolutional layer, we set padding mode to 'same' to keep number of time bins unchanged. The data is then fed into STT after normalization, activation and dropout layers.

## 3. Results

### *3.1. Transformer-based models perform better than CNN and are comparable to recurrent architectures*

We test 6 different deep models on 3 datasets. To qualitatively display the decoding ability of different decoders, we compare the decoder results with the real data from 3 datasets as shown in Figure 3. We also perform quantitative measurements of the decoder results. We use $R^2 = 1 - \frac{\sum_i (\hat{y}_i - y_i)^2}{\sum_i (y_i - \bar{y})^2}$ values to evaluate the goodness of fit, where $\hat{y}_i$ are the predicted values, $y_i$ are the true values, and $\bar{y}$ is the mean value. We use a ten-fold cross-validation method for training, using 90% of the data as training data and 10% of the data as test data. The final $R^2$ value is obtained by averaging the $R^2$ values across the x and y components of velocity or position of the test set for each fold. The one with the highest $R^2$ value is considered to have the best ability of decoding. The results are shown in Table 3.

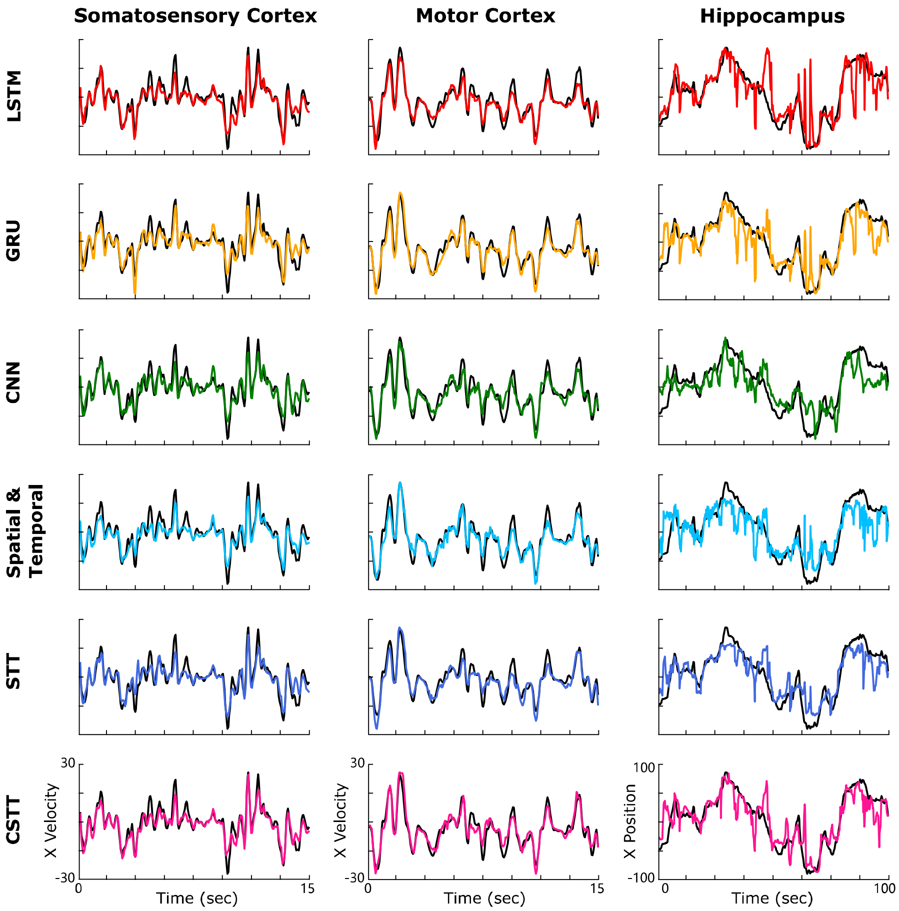

**Figure 3.** These are the decoder results clips, which reflect the goodness of fit. Results are from somatosensory cortex (left), motor cortex (middle) and hippocampus (right), for all 6 decoders. The black traces represent ground truth and the colored traces are the decoder results.

**Table 3.** The goodness of fit of six different models on the three datasets.

| Model | Somatosensory Cortex ( $R^2$ ) | Motor Cortex ( $R^2$ ) | Hippocampus ( $R^2$ ) |
|---|---|---|---|
| LSTM | 0.8600 | **0.8826** | **0.6088** |
| GRU | 0.8592 | 0.8787 | 0.5835 |
| CNN | 0.8498 | 0.8399 | 0.5034 |
| Spatial & Temporal | 0.8153 | 0.8208 | 0.5518 |
| STT | 0.8517 | 0.8644 | 0.5828 |
| CSTT | **0.8632** | 0.8734 | 0.5833 |

[1] Spatial & Temporal means a spatial Transformer followed by a temporal Transformer.

Our modified Transformer-based model STT achieves higher $R^2$ values than Spatial & Temporal Transformer on all the three datasets, which indicates that the combination of spatial and temporal attention does help improve model's decoding ability. The CSTT model performs better than STT due to convolutional structure which improves its modeling capacity. The CSTT model achieves the highest $R^2$ value of 0.8632 on the dataset collected from the somatosensory cortex. However, the LSTM maintains its dominance on the other two datasets.

Such difference across datasets may be due to various reasons, such as the size of the datasets or the length of the input sequences. The dataset from somatosensory cortex has the largest sample size of 61,339 with 13 bins per trial, while the volume of dataset from motor cortex and hippocampus is 25,299 and 22,283 with input length of 14 and 10 bins, respectively. We try to increase dataset size and sequence length to see how these two factors can affect the decoding ability of different models.

### *3.2. Transformer-based models perform better on large-scale datasets*

We use GLM model described in section 2.2 to generate synthetic spikes and get three fake datasets. The size of synthetic dataset from somatosensory cortex is ten times larger than the original with 613,390 trials. To achieve a similar size, we expand the hippocampus dataset to 30 times its original size, with 668,490 trials. For the dataset from motor cortex, we only expand it to 13 times the original size due to the limitation of computing device. We use them to do pre-training on LSTM, STT and CSTT respectively. Then train and test with real data. We compare the $R^2$ values of the three models before and after the dataset augmentation and the results are shown in Table 4.

**Table 4.** Comparison of decoding ability of LSTM, STT and CSTT on three datasets before and after expansion.

| Dataset | Model | Before | After |
|---|---|---|---|
| Somatosensory Cortex ( $R^2$ ) | LSTM | 0.8600 | 0.8650 |
| | STT | 0.8517 | 0.8644 |
| | CSTT | 0.8632 | **0.8734** |
| Motor Cortex ( $R^2$ ) | LSTM | 0.8826 | 0.8831 |
| | STT | 0.8644 | 0.8711 |
| | CSTT | 0.8734 | **0.8847** |
| Hippocampus ( $R^2$ ) | LSTM | 0.6088 | 0.6312 |
| | STT | 0.5828 | **0.6429** |
| | CSTT | 0.5833 | 0.6427 |

After data augmentation, the decoding performances of the three models improve on all the three datasets, with the two Transformer-based models showing a more significant improvement than the LSTM. The $R^2$ values of STT improve by 0.0127, 0.0067 and 0.0601 on the three datasets, and those of CSTT improve by 0.0102, 0.0113 and 0.0594, while those of LSTM only improve by 0.005, 0.0005 and 0.0224. Moreover, after data augmentation, CSTT achieves the best decoding results for both somatosensory cortex and motor cortex tasks, which are 0.8734 and 0.8834, respectively. STT scores the highest $R^2$ value of 0.6429 on hippocampus. Such improvements in decoding ability are consistent with our expectation that the Transformer-based models perform better on large-scale datasets.

### *3.3. Transformer-based models can better handle long-range dependencies*

In the above experiments we use 650 ms surrounding the movement (the concurrent bin, 6 bins before, and 6 bins after) for data from somatosensory cortex, 700 ms of neural activity (the concurrent bin and 13 bins before) for data from motor cortex and 2 s of surrounding neural activity (the concurrent bin, 4 bins before, and 5 bins after) for data from hippocampus. We try to increase the length of the input sequence to see how input length can influence models' decoding performances. Each time we add 10 bins to the input sequence (5 bins before and 5 bins after) and repeat 10 times. The results are shown in Figure 3.

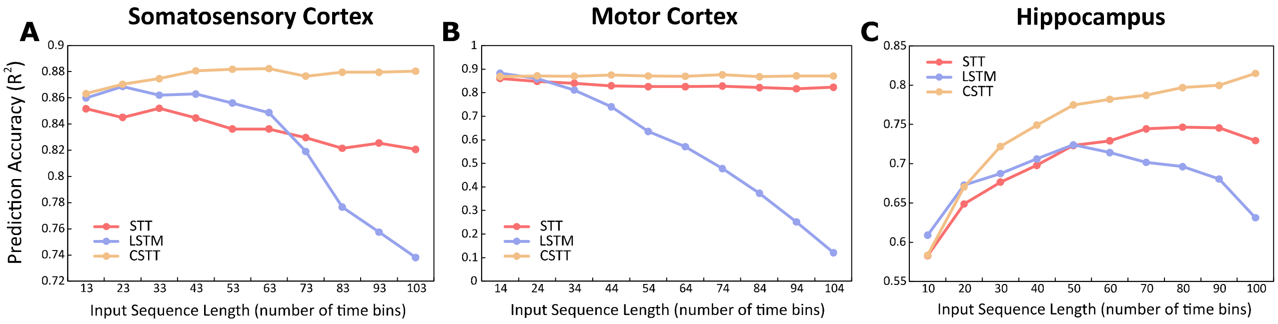

**Figure 4.** This is how the prediction accuracy varies with the input length. We increased input sequence length to evaluate its effect on prediction accuracy of STT, LSTM and CSTT on three datasets. Each column represents an experiment on a dataset.

The variation of prediction accuracy of the three models with increasing length of the input sequence is generally consistent across the three datasets. On the somatosensory dataset, the prediction accuracy of the CSTT remained between 0.86 and 0.88 with the increase of the input length and the prediction accuracy of the STT remained stable between 0.82 and 0.86 though with a decreasing trend. The prediction accuracy of LSTM varies the most drastically with the input length, except for a slight improvement before 23 bins, after which it keeps decreasing to around 0.74 at 103 bins.

On the Motor Cortex dataset, the stability difference in prediction accuracy between the LSTM and the two Transformer-based models is more obvious. As the input length grows, the prediction accuracy of the LSTM decreases rapidly from nearly 0.9 at 14 bins to nearly 0.1 at 104 bins. The two Transformer-based models, however, have maintained high stability.

On the dataset from hippocampus, the prediction accuracy of LSTM increases until 50 bins, then keeps decreasing and is surpassed by STT. The prediction accuracy of both Transformer-based models continues to increase, with STT slightly decreasing at 100 bins. CSST maintains the highest prediction accuracy after 20 bins.

The high stability in decoding performance with the growth of sequence length shows that Transformer-based models have a stronger ability of handling long-range dependencies.

## 4. Discussion

We modify Transformer for spike signals and introduce the Spatial Temporal Transformer (STT) model. To further improve its decoding ability, we combine the convolutional structures with STT and introduce the Convolutional Spatial Temporal Transformer (CSTT) model. The decoding ability of these two Transformer-based models are comparable to the recurrent neural network structure represented by LSTM. Moreover, the two Transformer-based models show excellent ability of handling large-scale and long-range dependencies. By increasing the dataset volume, our models achieve significant improvement in prediction accuracy, with CSTT outperforming the LSTM in decoding tasks for all three datasets and achieving the best decoding results in dataset from somatosensory cortex and motor cortex, while STT getting the highest $R^2$ value on dataset from hippocampus. In addition, Transformer-based models show stable and increased decoding performances when the input sequences are longer, while LSTM-based models deteriorate quickly. Our study of the inter-neuron connectivity also corroborates that neural signals do contain information generated by the interactions between neurons, which provides a clue for the design of neural decoding algorithms.

However, there are still many issues remained to be explored for the application of Transformer in neural decoding:

- We use GLM to generate synthetic data for data augmentation, which means an inevitable loss of features compared with real neural signals. The performances of Transformer on large-scale datasets with more abundant features can be further investigated in the future.
- Transformer's ability of dealing with long-range dependencies allows us to find appropriate input sequence length to achieve the best decoding results. Instead of manually setting the number of bins, we may let the model decide the best input length. And Transformer can offer the possibility to achieve a better decoding result compared with RNNs.
- We use a simple two-layer convolutional structure to form CSTT, which is proved to have better decoding ability than STT. Compared with the global feature extraction capability of Transformer, convolutional structure may have a stronger local feature extraction capability. It is a worthwhile problem to investigate how to combine CNN with Transformer better. In addition to convolutional structures, the combination of Transformer and other forms of feature extraction structures such as RNN or graph neural network is also worth further exploration.
- Transformer's attention structure is similar with convolutional structure in some way. There has been many successful examples of Transformer-based models being applied to images [11,12,19,20]. We may use Transformer structures on neural signals in the form of image such as fMRI and calcium image.

With the further development of neuroscience research, large-scale neural signal datasets will become available in the future. The superior processing capability of Transformer for large-scale datasets gives it potential to place RNNs as the most widely used deep learning algorithm in the field of neural decoding. Besides, there is now a boom of research on Transformer in machine learning community. As the research on Transformer deepens, we may achieve structures with stronger modeling ability and more applications under different scenarios. The possible explosion of neuronal data and research of Transformer in the future may bring huge change to neural decoding. Hence, our ability of decoding information from neuron population activities will be further improved, which will help us better understand neuron circuits and facilitate engineering applications.

**Funding:** This research received no external funding.

**Data Availability Statement:** Publicly available datasets were analyzed in this study. This public dataset can be downloaded from https://www.dropbox.com/sh/n4924ipcfjqc0t6/AACPWjxDKPEzQiXKUUFriFkJa?dl=0.

**Acknowledgments:** Thank Prof. Pengcheng Zhou for his guidance and encouragement.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Glaser, J.I.; Benjamin, A.S.; Chowdhury, R.H.; Perich, M.G.; Miller, L.E.; Kording, K.P. Machine Learning for Neural Decoding. *eNeuro* **2020**, *7*, ENEURO.0506-19.2020, doi:10.1523/ENEURO.0506-19.2020.
2. Ahmadi, N.; Constandinou, T.G.; Bouganis, C.-S. Decoding Hand Kinematics from Local Field Potentials Using Long Short-Term Memory (LSTM) Network. In Proceedings of the 2019 9th International IEEE/EMBS Conference on Neural Engineering (NER); March 2019; pp. 415–419.
3. Park, J.; Kim, S.-P. Estimation of Speed and Direction of Arm Movements from M1 Activity Using a Nonlinear Neural Decoder. In Proceedings of the 2019 7th International Winter Conference on Brain-Computer Interface (BCI); February 2019; pp. 1–4.
4. Naufel, S.; Glaser, J.I.; Kording, K.P.; Perreault, E.J.; Miller, L.E. A Muscle-Activity-Dependent Gain between Motor Cortex and EMG. *J. Neurophysiol.* **2019**, *121*, 61–73, doi:10.1152/jn.00329.2018.

5. Heelan, C.; Lee, J.; O'Shea, R.; Lynch, L.; Brandman, D.M.; Truccolo, W.; Nurmikko, A.V. Decoding Speech from Spike-Based Neural Population Recordings in Secondary Auditory Cortex of Non-Human Primates. *Commun. Biol.* **2019**, *2*, 1–12, doi:10.1038/s42003-019-0707-9.
6. Du, A.; Yang, S.; Liu, W.; Huang, H. Decoding ECoG Signal with Deep Learning Model Based on LSTM. In Proceedings of the TENCON 2018 - 2018 IEEE Region 10 Conference; October 2018; pp. 0430–0435.
7. Li, C.; Chan, D.C.W.; Yang, X.; Ke, Y.; Yung, W.-H. Prediction of Forelimb Reach Results From Motor Cortex Activities Based on Calcium Imaging and Deep Learning. *Front. Cell. Neurosci.* **2019**, *0*, doi:10.3389/fncel.2019.00088.
8. Petrosyan, A.; Lebedev, M.; Ossadtchi, A. Decoding Neural Signals with a Compact and Interpretable Convolutional Neural Network. *bioRxiv* **2020**, 2020.06.02.129114, doi:10.1101/2020.06.02.129114.
9. Dash, D.; Ferrari, P.; Wang, J. Decoding Imagined and Spoken Phrases From Non-Invasive Neural (MEG) Signals. *Front. Neurosci.* **2020**, *14*, 290, doi:10.3389/fnins.2020.00290.
10. Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding. *ArXiv181004805 Cs* **2019**.
11. Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; Zagoruyko, S. End-to-End Object Detection with Transformers. *ArXiv200512872 Cs* **2020**.
12. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An Image Is Worth 16x16 Words: Transformers for Image Recognition at Scale. *ArXiv201011929 Cs* **2021**.
13. Rao, R.; Bhattacharya, N.; Thomas, N.; Duan, Y.; Chen, X.; Canny, J.; Abbeel, P.; Song, Y.S. Evaluating Protein Transfer Learning with TAPE. *ArXiv190608230 Cs Q-Bio Stat* **2019**.
14. Glaser, J.I.; Perich, M.G.; Ramkumar, P.; Miller, L.E.; Kording, K.P. Population Coding of Conditional Probability Distributions in Dorsal Premotor Cortex. *Nat. Commun.* **2018**, *9*, 1788, doi:10.1038/s41467-018-04062-6.
15. Benjamin, A.S.; Fernandes, H.L.; Tomlinson, T.; Ramkumar, P.; VerSteeg, C.; Chowdhury, R.H.; Miller, L.E.; Kording, K.P. Modern Machine Learning as a Benchmark for Fitting Neural Responses. *Front. Comput. Neurosci.* **2018**, *12*, 56, doi:10.3389/fncom.2018.00056.
16. Mizuseki, K.; Sirota, A.; Pastalkova, E.; Buzsáki, G. Multi-Unit Recordings from the Rat Hippocampus Made during Open Field Foraging. 2009, 180 GB.
17. Mizuseki, K.; Sirota, A.; Pastalkova, E.; Buzsáki, G. Theta Oscillations Provide Temporal Windows for Local Circuit Computation in the Entorhinal-Hippocampal Loop. *Neuron* **2009**, *64*, 267–280, doi:10.1016/j.neuron.2009.08.037.
18. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention Is All You Need. *ArXiv170603762 Cs* **2017**.
19. Zheng, S.; Lu, J.; Zhao, H.; Zhu, X.; Luo, Z.; Wang, Y.; Fu, Y.; Feng, J.; Xiang, T.; Torr, P.H.S.; et al. Rethinking Semantic Segmentation from a Sequence-to-Sequence Perspective with Transformers. *ArXiv201215840 Cs* **2021**.
20. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin Transformer: Hierarchical Vision Transformer Using Shifted Windows. *ArXiv210314030 Cs* **2021**.