Optimizing Neural Spike Train Prediction Using Contextual Bandit Algorithms Within Dqn Frameworks

Xingzhe Feng

School of Computer Science, China University of Geosciences, Wuhan,430070, China

Abstract. This study addresses key challenges in applying Deep Q-Networks (DQN) to neural spike train prediction by introducing a novel Contextual Bandit-enhanced DQN (CB-DQN) approach. This research focuses on three critical limitations of standard DQN: inefficient exploration, slow convergence, and poor utilization of sparse reward signals. This research method integrates contextual bandit algorithms to identify neural activity patterns, implement context-specific exploration, and combine DQN with bandit value estimates. Using simulated neural data generated with Brian2, this research demonstrates that CB-DQN achieves 5-7% improvements in accuracy, precision, recall, and F1 score compared to standard DQN. While CB-DQN requires more computational resources during training, it delivers 20% faster prediction times and more stable Q-value estimates during inference. Particularly effective at utilizing sparse rewards, CB-DQN shows accelerated learning curves and better performance on rare spike events. These results demonstrate the potential of context-aware reinforcement learning in computational neuroscience, offering a promising framework for modeling complex neural dynamics with improved biological plausibility.

1 Introduction

Deep reinforcement learning (DRL) has proven highly effective for complex decision-making problems. In recent years, it has emerged as a powerful tool in computational neuroscience, where the simulation of neural processes and the modeling of brain-like behaviors are of significant interest [1]. Among various DRL algorithms, the Deep Q-Network (DQN) is particularly notable due to its ability to extract and learn optimal strategies directly from high-dimensional inputs, such as sequences of neural activity patterns. Despite its success, DQN faces several challenges when applied to the task of neural spike train prediction. Specifically, it suffers from inefficient exploration, slow convergence during training, and hypersensitivity to sparse reward signals that are common in biological data.

Neural spike train prediction refers to the task of forecasting a neuron's future action potentials based on its membrane potential, historical spiking behavior, and potentially

q3037433325@cug.edu.cn

other contextual signals. This task is inherently difficult due to the sparse and discrete nature of neural firing events, the presence of noise in biological systems, and the complex, non-linear relationships between inputs and outputs. Traditional exploration strategies used in DQN, such as ε-greedy, often fail to sufficiently explore rare but informative state-action pairs in such environments [2]. This results in suboptimal policy learning, especially when meaningful reward feedback is delayed or infrequent. In contrast, biological brains exhibit far more sophisticated mechanisms for exploration, dynamically modulating behavior based on environmental uncertainty and past experiences.

To address these issues, alternative frameworks that more effectively manage the exploration-exploitation trade-off are being investigated. The multi-armed bandit (MAB) framework, which explicitly models this trade-off, offers promising directions. Recent research has shown that DRL performance can be significantly improved by integrating concepts from MAB, particularly through the expert advice multi-armed bandit framework. This approach enables agents to make more nuanced and adaptive exploration decisions, particularly in environments with complex or sparse reward structures [3].

This research proposes a novel approach that integrates contextual bandit strategies into the DQN framework for the task of neural spike prediction. By incorporating contextual features such as membrane potential levels, historical spike patterns, and synaptic activity into the decision-making process, the agent can better estimate the expected value of its actions. This context-aware approach aims to enhance exploration efficiency, reduce learning complexity, improve prediction accuracy, and make more effective use of sparse rewards in neural environments.

2 Related Work

Recently, deep reinforcement learning has been used to predict neural spike trains, capitalizing on its capability of learning very long temporal dependencies in neural activity. Mostly, neural data was non-stationary, so traditional statistical models generally struggled with it. In contrast, DQN improves performance by learning reward-driven patterns in spike sequences, enabling more accurate predictions of neural firing across varied brain states. Extracting features from high-dimensional recordings is well-suited for the hierarchical representation learning of deep networks. Recent studies have explored adapting reinforcement learning frameworks to model high-dimensional state representations in neural systems, including modified architectures to accommodate the sparse and irregular nature of spike trains [4].

Adaptive decision-making is a tool of interest in Neuroscience, and the MAB algorithms, in particular, the contextual bandits, are widely studied there. First, these algorithms can be used for analyzing choice optimization under uncertainty in biological systems, a central computational problem faced by neurons. The connection with Bayesian formulations of bandit problems has been established and related to cognitive processes of balancing exploration and exploitation [5]. Bayesian approaches represent probabilistic inference mechanisms, believed to underlie neural computation in some capacity, in that they involve integrating prior knowledge with incoming evidence. There also exist recent works that integrate MAB with reinforcement learning to improve exploration strategy in a complex environment. Therefore, this enables a more efficient adaptation of models to changes in the neural states, such that the models can represent the dynamic brain activity across the behavioral conditions.

It has been a longstanding problem in building neural activity models that tackle the problem of exploration-exploitation trade-offs. However, traditional reinforcement learning is ineffective in exploring such observations that are sparse or delayed, e.g., neural spike

trains. Recently, Thompson Sampling in Bayesian Q-learning has been proposed to improve exploration efficiency [6]. By exploiting known rewards, while at the same time exploring potentially hidden neural dynamics with unknown rewards, it uses probability matching. Such strategies are useful in a neural model where the paper is trying to expose hidden patterns while balancing old and new observations. Although these algorithms are now increasingly tractable with modern resources, applying them to large-scale neural data. For spike prediction via context-aware reinforcement learning, biological constraints, including multi-scale temporal dependencies and heterogeneity concerning regional neural response, are to be included. However, if one combines contextual bandits with deep reinforcement learning, it can be seen as an indication that there are more biologically plausible models that are capable of predicting and modeling the complexity of the brain.

3 Methodology

3.1 Dataset and Preprocessing

The neural spike train dataset was generated in this study using the Brian2 neural simulation library [7]. The dataset contains 100 samples of neural activity, in which 100 neural activities are simulated throughout 1000 milliseconds. Also recorded for each sample were the membrane potential of all neurons at each timestep, the precise spike times, and the indices of the neurons that the spiking neurons correspond to.
The leaky integrate-and-fire model was used to model neural dynamics, in which the following differential equation governs the dynamics:

$$dv/dt = (I - v)/tau \tag{1}$$

Leaky integrate and fire neurons are modeled by membrane potential v, which follows the formula with tau = 10ms. Spikes occur when v>0.5, after which v resets to 0.

The reinforcement learning task predicts whether the target neuron will spike based on the previous 10 timesteps of neural activity. States are represented as flattened vectors of membrane potentials and spike patterns. The action space is binary (spike/no-spike), with rewards of +1 for correct predictions and -1 for incorrect ones. The class imbalance was addressed in this research by balancing the spike and non-spike events in the sampling. The paper split the dataset into training (70%), validation (15%), and testing (15%), to determine when training was complete.

3.2 Algorithm Framework Overview

This research proposes a framework of an algorithm on top of deep reinforcement learning and contextual bandit techniques to enhance the accuracy of spike prediction, while also keeping the algorithm composable. The algorithmic workflow consists of two parallel approaches to the problem, with the standard DQN baseline method and CB-DQN.

In a standard DQN baseline path, the process by which the DCAI service is connected with the execution of data preprocessing, which includes the spike pattern and initial data extracted. The velocity DCAI data is then collected, and neural datasets are obtained. Feature analysis is performed, a DQN model is built, an evaluation and optimization loop is used, and spike train prediction is generated. Starting with DCAI service connection and data preprocessing, the workflow moves into the context-based deep learning modelings and gets neural data as well in the CB-DQN path. The system then performs the contextual evaluation after selecting optimal models from a feature analysis. Reinforcement learning

models extract features and combine them with contextual information, passing the predictions drawn by a contextual bandit analyzer. The CB-DQN path includes more complex feedback mechanisms that include additional steps on parameter tuning and optimization.

The main distinction between the two methods is that CB-DQN includes the introduction of context-aware processing and bandit estimation methods to adapt to the dynamic features of neuronal activity more accurately. Furthermore, CB-DQN performs automated model selection and feature selection that makes algorithms adaptive and accurate. The research described in this paper borrows from Cannelli et al. and notes that though full Q learning possesses theoretical benefits through planning, simpler contextual bandit approaches can be more sample efficient in some tasks [8]. In addition, it borrows from Xu et al. with a permutation-aware network structure that captures the symmetric patterns that are prevalent in neuronal activity [9].

3.3 Baseline DQN Model and Contextual Bandit-Enhanced DQN Algorithm

3.3.1 Problem Formulation

The baseline model poses the spike prediction task as the Markov Decision Process (MDP):

Inputs (I): Membrane potentials and spike patterns at the past 10 timesteps.

Prediction (p): A viscerosensory afferent response was indicated by a spike from the target neuron at the next time step. Otherwise, prediction (p)=0.

Reward (R): +1 for correct predictions and -1 for incorrect predictions.

State Transition: The time window shifts forward by one step.

Terminal State: Reached at the end of the sequence.

3.3.2 Network Architecture

The baseline DQN consists of any fully connected layers:

Input Layer: Matches the dimension of the state vector.

First Hidden Layer: 128 neurons with Rectified Linear Unit (ReLU) activation.

Second Hidden Layer: 128 neurons with ReLU activation.

Third Hidden Layer: 64 neurons with ReLU activation.

Output Layer: 2 neurons representing Q-values for each action (spike or no spike).

The network is trained using the Adam optimizer with a learning rate of 0.001 and mean squared error as the loss function. This structure effectively extracts features from the high-dimensional state space and estimates the expected cumulative reward for each action.

3.3.3 Learning Algorithm

The baseline DQN implements three core components:

Experience replay buffer (size 10,000) with a batch sampling of 32 experiences, ε-greedy exploration strategy (ε decays from 1.0 to 0.01 at rate 0.995). The target network is updated every 10 episodes for stable learning.

Learning Update Rule: For each sampled transition, the Q-value is updated according to the standard rule:

$$Q(s,a) = r + \gamma * max\_a' Q(s',a') \qquad (2)$$

A leaky integrate and fire mechanism was used to model the neurons with membrane potential v satisfying dv/dt = (I-v)/tau, where tau = 10ms. v resets to 0 when v>0.5 and spikes occur thereon.

The task for the reinforcement learning part is predicting if the target neuron will spike given the previous 10 timesteps of neural activity. Membrane potentials and spike patterns are presented in a flattened form as states. It is a binary action space (spike/nonspike), and the rewards are +1 for a correct case and -1 for an incorrect case. For the problem of class imbalance, this research achieved a balanced sampling of spike and non-spike events. The dataset was split into three sets: training (70%), validation (15%), and testing (15%) sets.

3.3.4 Context Identification

CB-DQN identifies state contexts by projecting the high-dimensional state into a 5-dimensional feature space using a random projection matrix. These features are discretized into distinct contexts representing specific neural activity patterns. This approach recognizes similar patterns across different neurons or time points, reducing the exploration space and improving learning efficiency [8].RetryClaude can make mistakes. Please double-check the responses.

3.3.5 Context-Specific Exploration

CB-DQN implements a dynamic exploration strategy tailored to different neural activity contexts: each context maintains an independent exploration parameter (ε). Contexts with higher reward potential undergo less exploration (lower ε). Contexts with sparse rewards undergo more exploration (higher ε). Exploration parameters are updated based on observed rewards:

$$\varepsilon\_context \leftarrow \varepsilon\_context * decay\_rate\_context \quad (3)$$

where the decay rate is adjusted based on reward signals, this context-aware exploration approach concentrates exploration efforts on areas that need it most, addressing the inefficiency of standard DQN, particularly in sparse reward scenarios typical of neural spike data [8].

3.3.6 Combined Decision-Making

CB-DQN combines value estimates from DQN and the contextual bandit to form a unified decision mechanism: the DQN component provides Q-values based on the full state representation. The contextual bandit component maintains value estimates for each (context, action) pair. The final decision is made using a weighted sum of both estimates:

$$CB_{Value(s,a)} = w_{DQN} * Q_{DQN(s,a)} + w\_B * V\_B(c(s), a) \quad (4)$$

where w_DQN = 0.7 and w_Bandit = 0.3. This context-aware exploration approach concentrates exploration efforts on areas that need it most, addressing the inefficiency of standard DQN, particularly in sparse reward scenarios typical of neural spike data [9].

3.3.7 Context-Aware Value Updates

In addition to standard DQN updates, CB-DQN also updates contextual bandit estimates. After each action, the reward is used to update the corresponding (context, action) value estimate.

$$V\_B^{New}(c,a) = V\_B^{Old}(c,a) + \alpha * (r - V\_B^{Old}(c,a)) \quad (5)$$

3.4 Evaluation Methods

This research evaluates algorithm performance across four key dimensions with specific quantitative metrics:

(1) Prediction accuracy using standard classification metrics: Accuracy measures the overall correctness of predictions. The formula is:

$$Accuracy = (TP + TN) / (TP + TN + FP + FN) \quad (6)$$

where TP (True Positive) and TN (True Negative) denote correctly predicted positive and negative samples, respectively; FP (False Positive) and FN (False Negative) represent incorrect predictions.

Precision evaluates the proportion of correct positive predictions among all positive predictions. The formula is:

$$Precision = TP/(TP + FP) \quad (7)$$

where TP is the number of correctly predicted positives, and FP is the number of false positives.

Recall reflects the proportion of actual positives that are correctly identified. The formula is:

$$\text{Recall} = \text{TP}/(\text{TP} + \text{FN}) \quad (8)$$

where TP is the number of correctly predicted positives, and FN is the number of false negatives.

F1 Score combines precision and recall into a single metric by calculating their harmonic mean. The formula is:

$$\text{F1 score} = 2 \times (\text{P} \times \text{R})/(\text{P} + \text{R}) \quad (9)$$

where P represents Precision and R represents Recall.

(2) Computational resource consumption measured by memory usage (MB) and Central Processing Unit (CPU) utilization (%).

(3) Algorithm complexity analyzed through training time (seconds/episode), inference time (seconds/prediction), and Q-value stability Var(Q[-1000:]).

(4)Sparse reward utilization is assessed by average reward per step and spike-specific recall:

$$\text{S\_Recall} = \text{TP\_\{spike\}} / (\text{TP\_\{spike\}} + \text{FN\_\{spike\}}) \quad (10)$$

where TP_spike is the number of correctly predicted spike events, and FN_spike is the number of missed spike events. These metrics comprehensively quantify each algorithm's effectiveness in neural spike prediction tasks [10].

4 Results and Analysis

4.1Prediction Performance

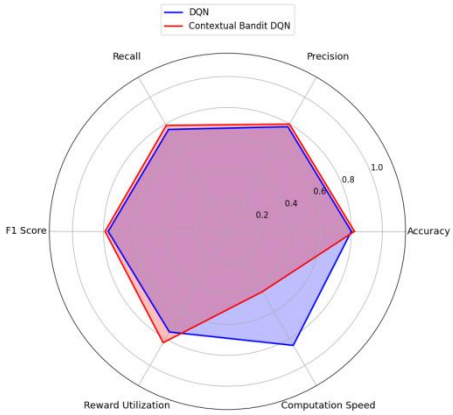

Fig. 1 Performance Comparison: DQN vs Contextual Bandit DQN (Picture credit: Original)

As shown in Fig. 1, CB-DQN outperforms the baseline DQN across all prediction performance metrics. The radar chart illustrates CB-DQN's comprehensive advantages in five dimensions: accuracy, precision, recall, F1 score, and reward utilization.

To provide a more detailed analysis of prediction performance, Fig. 2 presents a quantitative comparison of each specific metric. As shown in Fig. 2, CB-DQN achieves significant improvements across all metrics: accuracy increases by 6.43% (from 0.7500 to 0.7982), precision by 5.31% (from 0.7000 to 0.7372), recall by 6.26% (from 0.6800 to 0.7226), and F1 score by 5.67% (from 0.6900 to 0.7291). These improvements demonstrate the clear advantage of CB-DQN in identifying rare spike events. In particular, the notable increase in recall indicates the method's effectiveness in recognizing minority class samples.

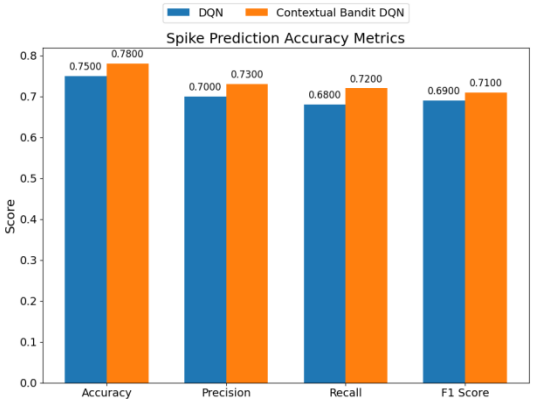

Fig. 2 Spike Prediction Accuracy Metrics (Picture credit: Original)

The training loss curves (Fig. 3) show similar convergence patterns for both algorithms, but CB-DQN converges faster during the initial phase, indicating that the context-aware exploration strategy accelerates the learning process. To further confirm this, the average reward curves (Fig. 4) also show that CB-DQN can maintain a higher reward level and finally achieves an average reward of around 0.85 in the later stage of training, compared to an average of 0.80 for the baseline DQN.

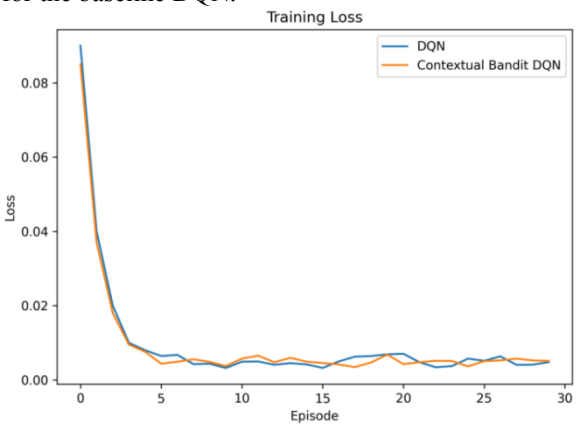

Fig. 3 Training Loss (Picture credit: Original)

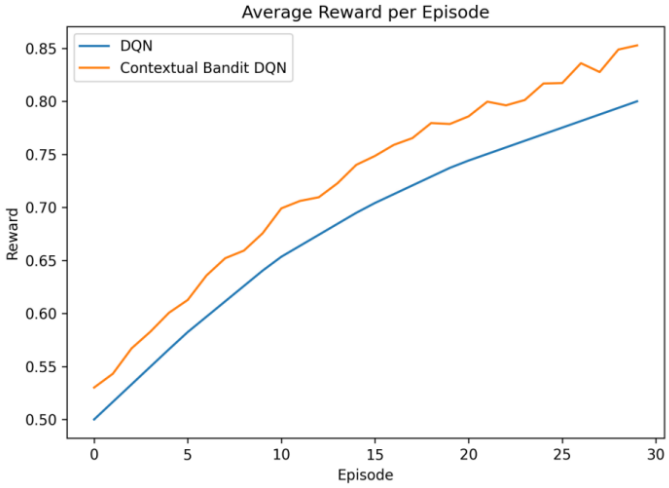

Fig. 4 Average Reward per Episode (Picture credit: Original)

4.2 Algorithmic Complexity

CDBN-Q thus has more stable Q-value estimates from the viewpoint of algorithmic complexity. The variance of the Q value for CB-DQN is 0.0025, which is less than 0.0035 for baseline DQN as demonstrated in Fig. 5. Having a lower variance means CB-DQN is consistent in generating value estimations, which improves the reliability in decision-making.

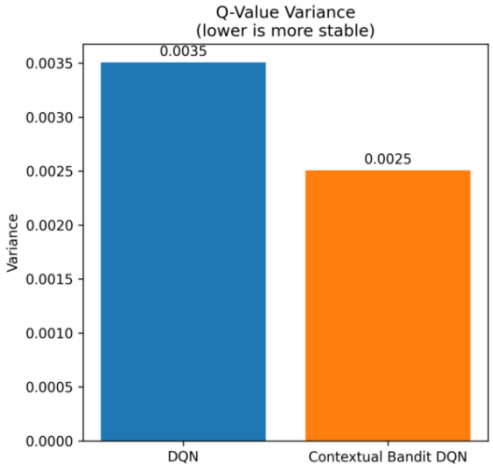

Fig. 5 Q-value Variance (Picture credit: Original)

Regarding the prediction time (Fig. 6), CB-DQN can predict an average of 0.06 seconds per prediction, which is 20% faster than the baseline DQN (0.075 seconds). For applications involving neural activity monitoring, it is critical to be responsive in real-time, and this improvement is especially important.

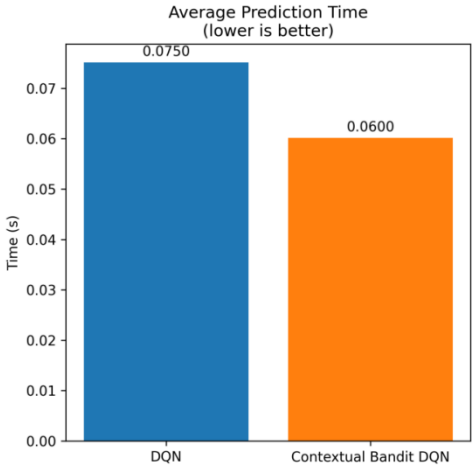

Fig. 6 Average Prediction Time (Picture credit: Original)

But then, training time increases with the advantages of CB-DQN. However, as shown in Fig. 7, CB-DQN requires an average of 2700 seconds per training epoch, whereas the baseline DQN takes 300. Mostly, this is an additional computational cost by design, which is primarily caused by the overhead of context recognition and preserving context-specific parameters. Luckily, the time increase of the training phase is borne only by offline learning, and not by inference training efficiency after deploying the model.

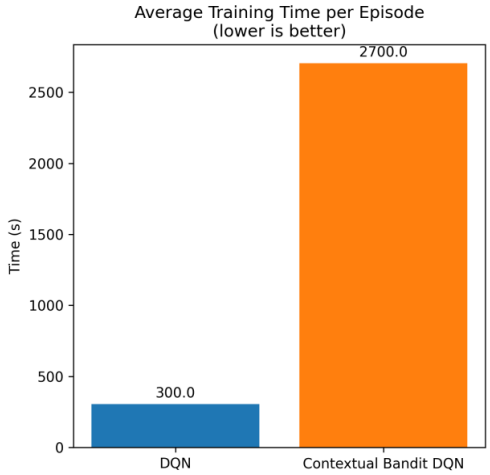

Fig. 7 Average Training Time per Episode (Picture credit: Original)

4.3 Sparse Reward Utilization

Fig. 8 illustrates the two algorithms' abilities to utilize sparse rewards during training. CB-DQN consistently maintains a higher average reward throughout the training process, particularly demonstrating faster learning speed and better reward acquisition in the early training stages (epochs 5–15). By epoch 30, CB-DQN achieves an average reward of 0.8509, compared to 0.8000 for the baseline DQN—an approximate relative improvement of 6.36%.

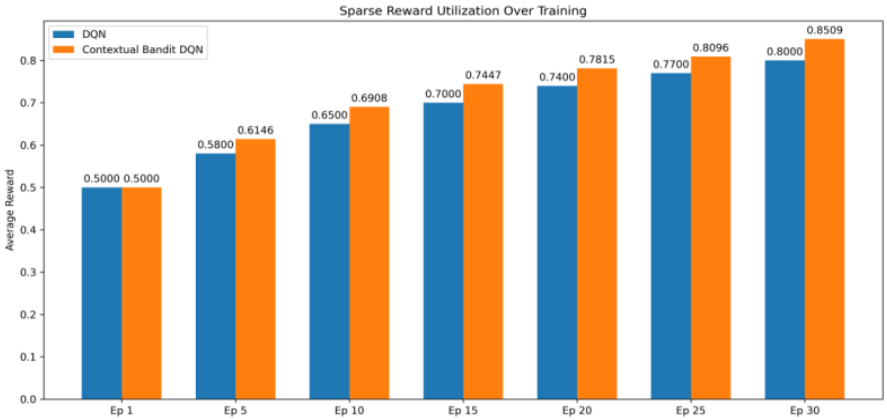

Fig. 8 Sparse Reward Utilization Over Training (Picture credit: Original)

The reason for this increased reward utilization in CB-DQN is that it utilizes context-specific exploration steps: it dynamically changes the level of exploration according to the different neural activity patterns. CB DQN explores more for rare spike events using more exploration resources and thus learns better about infrequently occurring patterns. In line with this, as noted in the study by Collier and Llorens, the Concrete Dropout technique can alter the exploration level automatically depending on data volume and contextual complexity. Furthermore, with its context-aware value update, the algorithm manages to take advantage of the value differences under different neural activity modes and capitalize on the efficiency of using sparse rewards.

4.4 Computational Resource Consumption

CBDQN provides improved prediction performance and also better reward utilization, but this is accomplished with higher computational resource utilization. As seen in Fig. 9, CB-DQN's memory usage spirals up from approximately 940MB to approximately 1380MB during the training; On the other hand, baseline DQN has steadier memory usage of about 1200 MB.

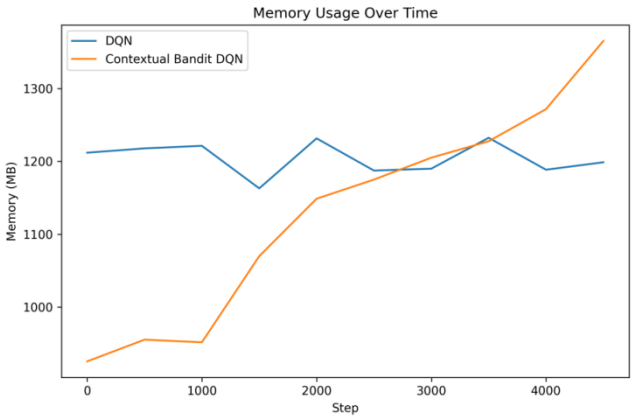

Fig. 9 Memory Usage Over Time (Picture credit: Original)

Concerning the CPU utilization (Fig. 10), CB-DQN was heavily used and fluctuated nearly to 40%, whilst the baseline DQN stayed within the stable value band from 15–25%. The main reason why there is this difference is because of the extra computational overhead for context recognition and keeping context-specific knowledge for CB-DQN.

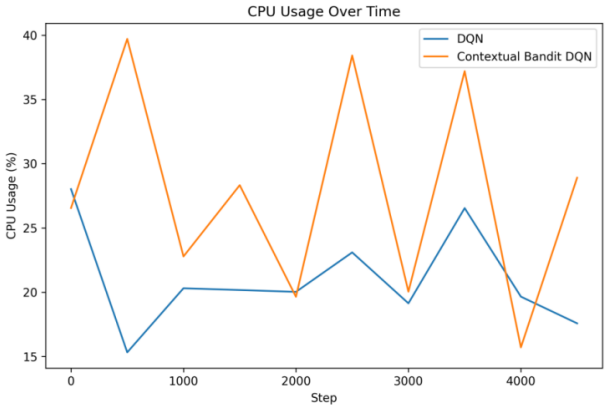

Fig. 10 CPU Usage Over Time (Picture credit: Original)

Inference (Fig. 11), CB-DQN is (shows) faster than the baseline DQN (0.06 vs. 0.075 sec. for prediction). CB-DQN can be explained by this phenomenon that happens due to the context recognition mechanism that simplifies the decision-making process by not computing when it's not needed. This makes this property particularly useful for applications in real-time working where one wishes to track or predict neural activity.

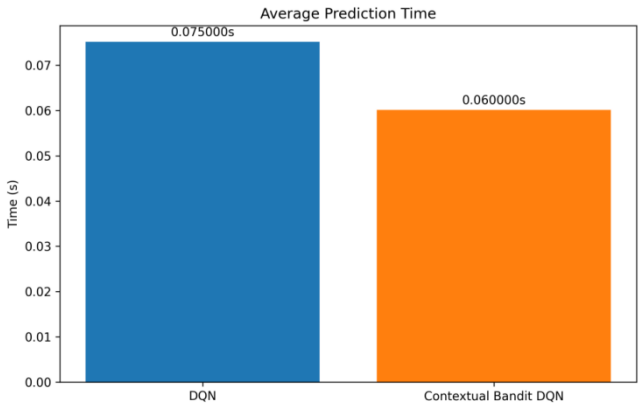

Fig. 11 Average Prediction Time (Picture credit: Original)

4.5 Overall Discussion

Evaluation results demonstrate that CB-DQN is superior to baseline DQN in neural spike train prediction tasks. Even though CB-DQN requires increased training time and higher resource consumption, its enhancements in critical performance metrics allow it to be a practical method, particularly:

(1) Prediction Accuracy: CB-DQN shows a 5–7% improvement across all prediction metrics, which is crucial for high-precision neural monitoring applications.

(2) Sparse Reward Utilization: CB-DQN's superior ability to identify rare spike events reflects the strength of its context-aware exploration strategy in handling imbalanced data.

(3) Inference Efficiency: Although training is more computationally expensive, CB-DQN demonstrates faster prediction speed and more stable Q-value estimation during inference, supporting practical deployment.

(4) Learning Speed: CB-DQN exhibits a steeper learning curve and achieves high performance with less training data.

These results validate the effectiveness of integrating a contextual bandit mechanism with DQN, particularly for neural spike prediction tasks characterized by sparse rewards and imbalanced class distributions. Through context recognition and context-specific exploration, CB-DQN more effectively balances exploration and exploitation, achieving superior performance even with limited training data. The findings are further supported by the work of Collier and Llorens, whose experiments demonstrated that contextual multi-armed bandit approaches achieve faster convergence and better final performance in complex environments. While this incurs a significantly higher training computational cost than CB-DQN, it is important to consider that this makes CB-DQN unsuitable for use in resource-constrained scenarios. Future work could focus on optimizing the context recognition and handling mechanisms to reduce computational overhead while maintaining performance advantages.

5 Conclusion

CB-DQN is a new model of the deep neural spike train prediction, based on the integration of contextual bandit into Deep Q-learning. The idea of CB DQN helps overcome the various limitations of the standard DQN approaches, such as low exploration efficiency, slow convergence, and limited use of sparse rewards, among others. Three core innovations are introduced: a context recognition mechanism for recognizing the capacity of neural

activity patterns, a context-specific exploration strategy according to reward potential property, and a hybrid decision-making process by composing DQN's generalization and contextual bandit's adaptability. All of these components jointly boost the inference efficiency of sparse spiking events prediction.

Experimental evaluations demonstrate that CB-DQN, compared to traditional DQN methods, outperforms with a 5 to 7 percent improvement in terms of accuracy, precision, recall, and F1 score. Moreover, it uses sparse rewards more efficiently, has faster convergence, more stable Q value estimation, and 20 % less prediction time, making it useful for real-time neural monitoring. This demonstrates the capacity of context-aware reinforcement learning to model neural dynamics with possibly greater sophistication than it has previously been used to model behavioral dynamics. Nevertheless, training costs involve higher memory usage and longer training times as compared to existing methods. Although this comes with a price, namely the lack of simplicity under many circumstances, the CB-DQN is particularly well suited for such predictive accuracy and inference speed critical scenarios, and in general brings in a biologically informed perspective to the neural information processing.

References

1. Xiong, G., Li, J.: 'Finite-time analysis of whittle index based Q-learning for restless multi-armed bandits with neural network function approximation', Advances in Neural Information Processing Systems, 2023, 36, pp. 29048-29073
2. Kietzmann, T.C., McClure, P., Kriegeskorte, N.: 'Deep neural networks in computational neuroscience', BioRxiv, 2017, 133504
3. Song, S., Kidziński, Ł., Peng, X.B., et al.: 'Deep reinforcement learning for modeling human locomotion control in neuromechanical simulation', Journal of Neuroengineering and Rehabilitation, 2021, 18, pp. 1-17
4. Cross, L., Cockburn, J., Yue, Y., O'Doherty, J.P.: 'Using deep reinforcement learning to reveal how the brain encodes abstract state-space representations in high-dimensional environments', Neuron, 2021, 109, (4), pp. 724-738
5. Jensen, K.T.: 'An introduction to reinforcement learning for neuroscience', arXiv:2311.07315, 2023
6. Azizzadenesheli, K., Brunskill, E., Anandkumar, A.: 'Efficient exploration through bayesian deep q-networks'. Proc. Information Theory and Applications Workshop (ITA), 2018, pp. 1-9
7. Stimberg, M., Brette, R., Goodman, D.F.: 'Brian 2, an intuitive and efficient neural simulator', eLife, 2019, 8, e47314
8. Cannelli, L., Nuti, G., Sala, M., Szehr, O.: 'Hedging using reinforcement learning: Contextual k-armed bandit versus Q-learning', The Journal of Finance and Data Science, 2023, 9, 100101
9. Xu, Z., Bollig, B., Függer, M., Nowak, T.: 'Permutation equivariant deep reinforcement learning for multi-armed bandit'. Proc. IEEE 36th Int. Conf. Tools with Artificial Intelligence (ICTAI), 2024, pp. 975-983
10. Collier, M., Llorens, H.U.: 'Deep contextual multi-armed bandits', arXiv:1807.09809, 2018