

Neural Exploitation and Exploration of Contextual Bandits

Yikun Ban

UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

YIKUNB2@ILLINOIS.EDU

Yuchen Yan

UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

YUCHENY5@ILLINOIS.EDU

Arindam Banerjee

UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

ARINDAMB@ILLINOIS.EDU

Jingrui He

UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

JINGRUI@ILLINOIS.EDU

Editor: Chris Wiggins

Abstract

In this paper, we study the neural exploration strategy for contextual bandits. The dilemma of exploitation and exploration widely exists in real-world applications such as recommender systems, online advertising, and clinical trials. Contextual bandits provide principled methods to solve this dilemma, including two prevalent techniques: Thompson Sampling (TS), and Upper Confidence Bound (UCB). Neural contextual bandits have been studied to adapt to the non-linear reward function, combined with TS or UCB strategies for exploration. In this paper, we introduce, EE-Net, which is a novel framework to utilize another neural network to learn the potential gain of exploitation neural network for exploration, different from UCB-based and TS-based approaches that rely on the large-deviation-based statistical confidence bound. In addition, we provide an instance-based $\tilde{O}(\sqrt{T})$ regret upper bound for EE-Net with a new proof workflow. Empirically, we show that EE-Net outperforms related linear and neural contextual bandit baselines on real-world datasets.

Keywords: Multi-armed Bandits, Neural Contextual Bandits, Neural Networks, Exploitation and Exploration, Regret Analysis

1. Introduction

The stochastic contextual multi-armed bandit (MAB) (Lattimore and Szepesvári, 2020) has been extensively studied in the machine learning community for decades as a solution to sequential decision-making problems. This framework has practical applications in various fields, such as online advertising (Li et al., 2010) and personalized recommendation (Ban et al., 2024b; Ban and He, 2021b). In the standard contextual bandit setting, a learner is presented with a set of n arms in each round, where each arm is characterized by a context vector. The learner then selects and plays one arm based on a specific strategy, receiving a corresponding reward. The objective is to maximize the cumulative rewards over T rounds.

MAB provides principled solutions for the trade-off between Exploitation and Exploration (EE). On one hand, the learner should exploit the information from the collected data; on the other hand, the learner should explore the under-explored arms to obtain new information. The widely-used strategies for EE trade-off includes the following three techniques: Epsilon-

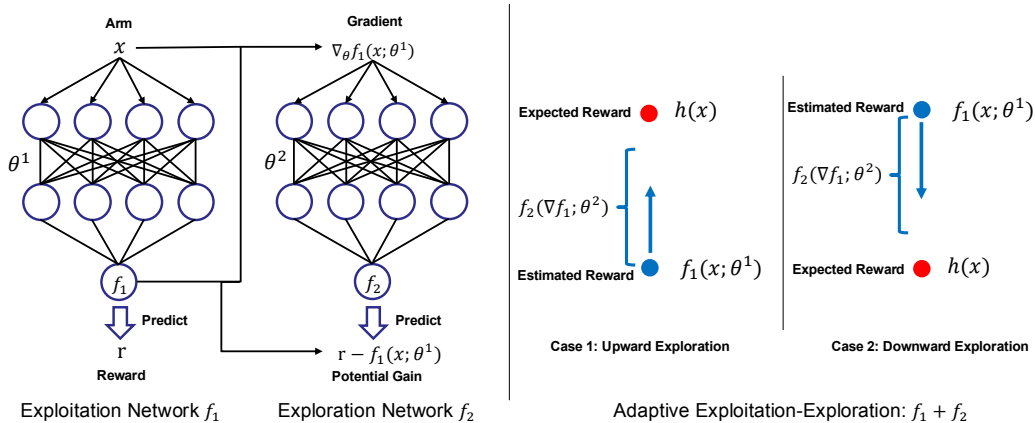


Figure 1: Exploration direction (right side): (1) "Upward" exploration should be performed when the model underestimates the arm’s reward; (2) "Downward" exploration should be performed when the model overestimates the arm’s reward. EE-Net (the proposed strategy), depicted in the left side, intends to adaptively make exploration according to the estimated potential gain of arm.

greedy (Langford and Zhang, 2008), Thompson Sampling (TS) (Thompson, 1933; Kveton et al., 2021), and Upper Confidence Bound (UCB) (Auer, 2002; Ban and He, 2020). Based on the realizability assumptions on reward function, the first mainstream is the linear contextual bandits, where the reward is assumed to be a linear function with respect to arm context vectors. Linear bandits have been well studied and succeeded both empirically and theoretically (Bouneffouf et al., 2020; Slivkins et al., 2019). UCB-based algorithms (Li et al., 2010; Chu et al., 2011; Wu et al., 2016; Ban and He, 2021b) calculate the confidence ellipsoid of estimated reward based on ridge regression. The selection criterion is to pull the arm with the maximal upper confidence bound concerning the estimated reward. TS-based algorithms (Agrawal and Goyal, 2013; Abeille and Lazaric, 2017; Osband et al., 2023a) formulate each arm with a prior distribution and select the arm with the maximal posterior probability of being the best arm (Valko et al., 2013). Another mainstream, neural contextual bandit, has gained attention in recent years. Neural bandits are able to utilize deep neural networks to learn the underlying non-linear reward functions, thanks to the powerful representation ability. Considering the past selected arms and received rewards as training samples, a neural network is built for exploitation. Zhou et al. (2020) compute a gradient-based upper confidence bound and use the UCB strategy to select arms. Zhang et al. (2021) formulate each arm as a normal distribution and the standard deviation is calculated based on the gradient of exploitation neural network, and then uses the TS strategy to choose arms.

Figure 1 illustrates the primary motivation for this work by examining the exploration mechanism in detail. Let $h(\mathbf{x})$ represent the expected reward of an arm \mathbf{x} , and let $f_1(\mathbf{x}; \theta^1)$ denote the reward estimated by the exploitation model f_1 . The difference between the estimated reward and the expected reward divides exploration into two categories: "upward" exploration and "downward" exploration. Upward exploration occurs when the expected reward is greater than the estimated reward, i.e., $h(\mathbf{x}) > f_1(\mathbf{x}; \theta^1)$. This situation indicates that the learner’s model has underestimated the reward, necessitating an additional value to be added to the estimated reward to reduce the discrepancy between the expected and

estimated rewards. Conversely, downward exploration happens when the expected reward is less than the estimated reward, i.e., $h(\mathbf{x}) < f_1(\mathbf{x}; \boldsymbol{\theta}^1)$. This scenario suggests that the model has overestimated the reward, thus requiring a value to be subtracted from the estimated reward to minimize the gap.

However, existing exploration strategies may not effectively handle both types of exploration. Specifically, UCB-based approaches (e.g., (Zhou et al., 2020)) select an arm based on the estimated reward $f_1(\mathbf{x}; \boldsymbol{\theta}^1)$ plus its confidence bound radius (a positive value). As a result, these approaches may perform poorly when $f_1(\mathbf{x}; \boldsymbol{\theta}^1)$ overestimates $h(\mathbf{x})$ (downward exploration required). On the other hand, TS-based methods (e.g., Wang et al. (2021b)) select arms based on a reward sampled from a distribution where the mean is $f_1(\mathbf{x}; \boldsymbol{\theta}^1)$. However, this distribution is not dynamically adapted for upward or downward exploration. Finally, epsilon-greedy algorithms (e.g., Auer (2002)) randomly select arms with a certain probability, making them unable to directly adjust to the upward or downward exploration.

In this paper, we propose a novel neural exploration strategy named "EE-Net", to adaptively perform upward and downward exploration. Similar to other neural bandits, EE-Net has an exploitation network f_1 to estimate the reward for each arm by exploiting the collected data. The crucial difference from existing works is that EE-Net has an exploration network f_2 that leverages the underlying exploitation information of f_1 to learn the "potential gain" of an arm compared to its current estimated reward. The input of the exploration network f_2 is the gradient of f_1 to incorporate the information of arm contexts and the discriminative ability of f_1 . The ground truth for training f_2 is the potential gain, which is the residual between the observed reward and the estimated reward by f_1 . This potential gain effectively indicates the direction for exploration, guiding f_2 to make upward or downward exploration. Ultimately, the two neural networks, f_1 and f_2 , work together to select the arms. Figure 1 depicts the architecture of EE-Net. To sum up, the contributions of this paper can be summarized as follows:

1. We propose a novel neural exploration strategy in contextual bandits, EE-Net, where another neural network is assigned to learn the potential gain compared to the current reward estimate for adaptive exploration, in addition to the neural network that exploits the collected data for exploitation.
2. Under mild assumptions of over-parameterized neural networks, we provide an instance-based $\tilde{O}(\sqrt{T})$ regret upper bound of for EE-Net, where the complexity term in this bound is easier to interpret, and this bound is at least as good as the existing works.
3. We conduct experiments on real-world datasets, showing that EE-Net outperforms baselines, including linear and neural versions of ϵ -greedy, TS, and UCB.

Next, we discuss the problem definition in Sec.3, elaborate on the proposed EE-Net in Sec.4, and present our theoretical analysis in Sec.5. In the end, we provide the empirical evaluation (Sec.6) and conclusion (Sec.7).

2. Related Work

Assuming linear reward realization, linear UCB-based bandit algorithms (Abbasi-Yadkori et al., 2011; Li et al., 2016) and linear Thompson Sampling (Agrawal and Goyal, 2013;

Abeille and Lazaric, 2017) demonstrate impressive empirical performance across various real-world scenarios, achieving a near-optimal regret bound of $\tilde{O}(\sqrt{T})$. To relax the linearity assumption, Filippi et al. (2010) generalize the reward function to include both linear and non-linear components, adopting a UCB-based algorithm for estimation. Similarly, Bubeck et al. (2011) impose the Lipschitz property on the reward metric space and develop a hierarchical optimization approach for selection. Valko et al. (2013) embed the reward function into a Reproducing Kernel Hilbert Space and propose kernelized TS/UCB bandit algorithms.

To learn non-linear reward functions, deep neural networks have been adapted to bandits in various ways. Riquelme et al. (2018); Lu and Van Roy (2017) construct L -layer DNNs to learn arm embeddings and apply Thompson Sampling on the last layer for exploration. Zhou et al. (2020) introduce a provable neural-based contextual bandit algorithm with a UCB exploration strategy, which Zhang et al. (2021) extend to the TS framework. Their regret analysis leverages recent advances in the convergence theory of over-parameterized neural networks (Du et al., 2019; Allen-Zhu et al., 2019) and employ the Neural Tangent Kernel (Jacot et al., 2018; Arora et al., 2019) to establish connections with linear contextual bandits (Abbasi-Yadkori et al., 2011). Ban and He (2021a) further integrate convolutional neural networks with UCB exploration for visual-aware applications. Xu et al. (2020) utilize UCB-based exploration on the last layer of neural networks to reduce computational costs associated with gradient-based UCB. Qi et al. (2022) explore the correlation among arms in contextual bandits. Ban et al. (2021) introduce a multi-facet bandit problem where one bandit formulates one facet, respectively. Unlike the aforementioned works, EE-Net retains the powerful representation capability of neural networks to learn the reward function and, for the first time, employs a separate neural network for exploration.

Recently, more ideas have been proposed by other neural bandit works (Ban et al., 2022a; Kassraie and Krause, 2022; Dai et al., 2022; Gu et al., 2024; Hwang et al., 2023; Lin et al., 2023), and tailored for various application scenarios such as active learning (Wang et al., 2021a; Ban et al., 2022b, 2024a), meta-learning (Ban et al., 2024c; Qi et al., 2024b), and bandit-based graph learning (Qi et al., 2022, 2023; Kassraie et al., 2022). Ban et al. (2024d) studies PageRank-based bandit approaches for link prediction in graphs. Deb et al. (2023) use inverse reward gap of neural approximation for exploration, and Jia et al. (2021) achieve efficient exploration by adding perturbations to received rewards in the training process. Qi et al. (2024a) studies neural contextual bandits in the presence of adversarial corruptions. (Osband et al., 2023b,a) propose a new method for approximating Thompson sampling using epistemic neural networks that are designed to produce accurate joint predictive distributions, which is further extended to the exploration in language models (Dwaracherla et al., 2024). Yang et al. (2026) identifies inherent bias in group-relative advantage estimators. Lu et al. (2026) studies contextual rollout bandits in reinforcement learning with verifiable rewards. Huang et al. (2025) studies bandit-based adaptive sample scheduling for direct preference optimization. Qi et al. (2026) studies bi-level hierarchical neural contextual bandits for online recommendation. To enhancing the exploration in reinforcement learning, (Burda et al., 2018) involves using two neural networks: a fixed, randomly initialized target network and a predictor network trained to mimic the target’s outputs. Then, the learner explore the states where the prediction error between these networks is high, whereas EE-net integrates both exploitation and exploration directly into its neural network model. (Osband et al., 2018; Dwaracherla et al., 2022) propose using ensemble methods augmented with fixed prior

functions or bootstrapping to quantify uncertainty and drive exploration in deep reinforcement learning, which differ from the exploitation–exploration neural networks in EE-Net.

For more variants of exploration strategies, GIOR (Kveton et al., 2019) explores by randomizing the history of rewards with pseudo-rewards, ensuring optimism in its bootstrap mean estimates. Bayes-UCB (Kaufmann et al., 2012), derived from the Bayesian index perspective, relies on posterior quantiles for exploration and has been shown to achieve asymptotic optimality. Top-Two sampling (Jourdan et al., 2022; Russo, 2016), as an adaptation of Thompson Sampling, randomizes between a leader and a challenger arm, aiming to balance exploration and exploitation for best-arm identification.

While these methods employ different mechanisms to encourage optimism or randomized exploration, they generally operate in a single or random “direction” of exploration. In contrast, our proposed EE-Net introduces a novel neural-based bi-directional exploration strategy, where the dual-network design allows EE-Net to capture exploration in both directions.

3. Problem Definition

Under the setting of stochastic contextual bandits, in t -th round, $t \in [T]$ where the sequence $[T] = [1, 2, \dots, T]$, the learner is presented with n arms represented by n context vectors, $\mathbf{X}_t = \{\mathbf{x}_{t,1}, \dots, \mathbf{x}_{t,n}\}$, $\mathbf{x}_{t,i} \in \mathbb{R}^d$, $\forall i \in [n]$. Then, the learner is compelled to select one arm $\mathbf{x}_{t,\hat{i}}$, $\hat{i} \in [n]$, and observe the corresponding reward $r_{t,\hat{i}}$. For each arm $\mathbf{x}_{t,i}$, $i \in [n]$, its reward $r_{t,i}$ is assumed to be governed by the function:

$$r_{t,i} = h(\mathbf{x}_{t,i}) + \eta_{t,i}, \quad (3.1)$$

where the *unknown* reward function $h(\mathbf{x}_{t,i})$ can be either linear or non-linear and $\eta_{t,i}$ is the noise term. Here, we assume the noise $\mathbb{E}[\eta_{t,i}] = 0$, while related works such as (Zhou et al., 2020; Ban et al., 2021; Zhang et al., 2021) assume $\eta_{t,i}$ is conditioned zero-mean sub-Gaussian noise. Finally, the pseudo *regret* of T rounds is defined as:

$$\mathbf{R}_T = \mathbb{E} \left[\sum_{t=1}^T (r_{t,i^*} - r_{t,\hat{i}}) \right], \quad (3.2)$$

where $i^* = \arg \max_{i \in [n]} h(\mathbf{x}_{t,i})$ is the index of an arm with the maximal expected reward in round t . The goal of this problem is to minimize \mathbf{R}_T by optimizing the selection policy.

Notation. We denote by $\{\mathbf{x}_\tau\}_{\tau=1}^t$ the sequence $(\mathbf{x}_1, \dots, \mathbf{x}_t)$. We use $\|v\|_2$ to denote the Euclidean norm for a vector v , and $\|\mathbf{W}\|_2$ and $\|\mathbf{W}\|_F$ to denote the spectral and Frobenius norm for a matrix \mathbf{W} . We use $\langle \cdot, \cdot \rangle$ to denote the standard inner product between two vectors or two matrices. We may use $\nabla_{\boldsymbol{\theta}_t^1} f_1(\mathbf{x}_{t,i})$ or $\nabla_{\boldsymbol{\theta}_t^1} f_1$ to represent the gradient $\nabla_{\boldsymbol{\theta}_t^1} f_1(\mathbf{x}_{t,i}; \boldsymbol{\theta}_t^1)$ for brevity. We use $\{\mathbf{x}_\tau, r_\tau\}_{\tau=1}^t$ to represent the collected data up to round t .

4. Proposed Method: EE-Net

EE-Net is composed of two neural networks to exploit the collected data and make exploration respectively. For convenience, denote the first neural network by $f_1(\cdot; \boldsymbol{\theta}^1)$, named by "exploitation network", and denote the second neural network by $f_2(\cdot; \boldsymbol{\theta}^2)$, named by

Algorithm 1 EE-Net

Input: f_1, f_2, T (number of rounds), η_1 (learning rate for f_1), η_2 (learning rate for f_2), ϕ (normalization operator, Remark 2)

- 1: Initialize $\boldsymbol{\theta}_0^1, \boldsymbol{\theta}_0^2; \mathcal{H}_0 = \emptyset$
- 2: **for** $t = 1, 2, \dots, T$ **do**
- 3: Observe n arms $\{\mathbf{x}_{t,1}, \dots, \mathbf{x}_{t,n}\}$
- 4: **for** each $i \in [n]$ **do**
- 5: Compute $f_1(\mathbf{x}_{t,i}; \boldsymbol{\theta}_{t-1}^1), f_2(\phi(\mathbf{x}_{t,i}); \boldsymbol{\theta}_{t-1}^2)$
- 6: **end for**
- 7: $\hat{i} = \arg \max_{i \in [n]} (f_1(\mathbf{x}_{t,i}; \boldsymbol{\theta}_{t-1}^1) + f_2(\phi(\mathbf{x}_{t,i}); \boldsymbol{\theta}_{t-1}^2))$
- 8: Play $\mathbf{x}_{t,\hat{i}}$ and observe reward $r_{t,\hat{i}}$
- 9: $\mathcal{L}_1 = \frac{1}{2} (f_1(\mathbf{x}_{t,\hat{i}}; \boldsymbol{\theta}_{t-1}^1) - r_{t,\hat{i}})^2$
- 10: $\boldsymbol{\theta}_t^1 = \boldsymbol{\theta}_{t-1}^1 - \eta_1 \nabla_{\boldsymbol{\theta}_{t-1}^1} \mathcal{L}_1$
- 11: $\mathcal{L}_2 = \frac{1}{2} (f_2(\phi(\mathbf{x}_{t,\hat{i}}); \boldsymbol{\theta}_{t-1}^2) - (r_{t,\hat{i}} - f_1(\mathbf{x}_{t,\hat{i}}; \boldsymbol{\theta}_{t-1}^1)))^2$
- 12: $\boldsymbol{\theta}_t^2 = \boldsymbol{\theta}_{t-1}^2 - \eta_2 \nabla_{\boldsymbol{\theta}_{t-1}^2} \mathcal{L}_2$
- 13: **end for**

"exploration network". The exploration network f_2 is the primary novel component of our proposed system. Table 1 lists the structure of EE-Net.

(1) Exploitation network. The exploitation network f_1 is a neural network to map arms to rewards. In round t , the network is represented as $f_1(\cdot; \boldsymbol{\theta}_{t-1}^1)$, where the superscript of $\boldsymbol{\theta}_{t-1}^1$ denotes the network's index, and the subscript indicates the round when f_1 's parameters were last updated. For an arm $\mathbf{x}_{t,i}$ where $i \in [n]$, $f_1(\mathbf{x}_{t,i}; \boldsymbol{\theta}_{t-1}^1)$ calculates the "exploitation score" for $\mathbf{x}_{t,i}$ based on historical data. Following a specific criterion, after selecting arm $\mathbf{x}_{t,\hat{i}}$, a reward $r_{t,\hat{i}}$ is obtained. Therefore, we can conduct stochastic gradient descent (SGD) to update $\boldsymbol{\theta}_{t-1}^1$ based on $(\mathbf{x}_{t,\hat{i}}, r_{t,\hat{i}})$ and denote the updated parameters by $\boldsymbol{\theta}_t^1$.

2) Exploration network. Our exploration strategy is inspired by existing UCB-based neural bandits (Zhou et al., 2020; Ban et al., 2021). Given an arm $\mathbf{x}_{t,i}$, with high probability, the following UCB form holds:

$$|h(\mathbf{x}_{t,i}) - f_1(\mathbf{x}_{t,i}; \boldsymbol{\theta}_{t-1}^1)| \leq \kappa(\nabla_{\boldsymbol{\theta}_{t-1}^1} f_1(\mathbf{x}_{t,i}; \boldsymbol{\theta}_{t-1}^1)), \quad (4.1)$$

where h is defined in Eq. (3.1) and κ is an upper confidence bound represented by a function with respect to the gradient $\nabla_{\boldsymbol{\theta}_{t-1}^1} f_1$ (see more details and discussions in Appendix A.4). Then we have the following definition.

Definition 1 (Potential Gain) *In round t , given an arm $\mathbf{x}_{t,i}$, we define $h(\mathbf{x}_{t,i}) - f_1(\mathbf{x}_{t,i}; \boldsymbol{\theta}_{t-1}^1)$ as the "expected potential gain" for $\mathbf{x}_{t,i}$ and $r_{t,i} - f_1(\mathbf{x}_{t,i}; \boldsymbol{\theta}_{t-1}^1)$ as the "potential gain" for $\mathbf{x}_{t,i}$.*

Let $y_{t,i} = r_{t,i} - f_1(\mathbf{x}_{t,i}; \boldsymbol{\theta}_{t-1}^1)$. When $y_{t,i} > 0$, the arm $\mathbf{x}_{t,i}$ has positive potential gain compared to the estimated reward $f_1(\mathbf{x}_{t,i}; \boldsymbol{\theta}_{t-1}^1)$. A large positive $y_{t,i}$ makes the arm more suitable for upward exploration while a large negative $y_{t,i}$ makes the arm more suitable for

downward exploration. In contrast, $y_{t,i}$ with a small absolute value makes the arm unsuitable for exploration. Recall that traditional approaches such as UCB intend to estimate such potential gain $y_{t,i}$ using standard tools, e.g., Markov inequality and Hoeffding bounds from large deviation bounds.

Instead of calculating a large-deviation based statistical bound for $y_{t,i}$, we use a neural network $f_2(\cdot; \boldsymbol{\theta}^2)$ to represent g , where the input is $\nabla_{\boldsymbol{\theta}_{t-1}^1} f_1(\mathbf{x}_{t,i})$ and the ground truth is $r_{t,i} - f_1(\mathbf{x}_{t,i}; \boldsymbol{\theta}_{t-1}^1)$. Adopting gradient $\nabla_{\boldsymbol{\theta}_{t-1}^1} f_1(\mathbf{x}_{t,i})$ as the input is due to the fact that it incorporates two aspects of information: the features of the arm and the discriminative information of f_1 .

Moreover, in the upper bound of NeuralUCB (Zhou et al., 2020) or the variance of NeuralTS (Zhang et al., 2021), there is a recursive term $\mathbf{A}_{t-1} = \mathbf{I} + \sum_{\tau=1}^{t-1} \nabla_{\boldsymbol{\theta}_{\tau-1}^1} f_1(\mathbf{x}_{\tau,i}) \nabla_{\boldsymbol{\theta}_{\tau-1}^1} f_1(\mathbf{x}_{\tau,i})^\top$ which is a function of past gradients up to $(t-1)$ and incorporates relevant historical information. On the contrary, in EE-Net, the recursive term which depends on past gradients is $\boldsymbol{\theta}_{t-1}^2$ in the exploration network f_2 because we have conducted gradient descent for $\boldsymbol{\theta}_{t-1}^2$ based on $\{\nabla_{\boldsymbol{\theta}_{\tau-1}^1} f_1(\mathbf{x}_{\tau,i})\}_{\tau=1}^{t-1}$. Therefore, this form $\boldsymbol{\theta}_{t-1}^2$ is similar to \mathbf{A}_{t-1} in neuralUCB/TS, but EE-net does not (need to) make a specific assumption about the functional form of past gradients, and it is also more memory-efficient.

To summarize, in round t , we define $f_2(\nabla_{\boldsymbol{\theta}_{t-1}^1} f_1(\mathbf{x}_{t,i}); \boldsymbol{\theta}_{t-1}^2)$ as the "exploration score" for $\mathbf{x}_{t,i}$ to facilitate adaptive exploration. This score reflects the potential gain of $\mathbf{x}_{t,i}$ compared to our current exploitation score $f_1(\mathbf{x}_{t,i}; \boldsymbol{\theta}_{t-1}^1)$. After receiving the reward r_t , we can update $\boldsymbol{\theta}^2$ using gradient descent based on the collected training samples $\{\nabla_{\boldsymbol{\theta}_{\tau-1}^1} f_1(\mathbf{x}_{\tau,i}), r_\tau - f_1(\mathbf{x}_{\tau,i}; \boldsymbol{\theta}_{\tau-1}^1)\}_{\tau=1}^t$. Additionally, we propose two heuristic forms for f_2 's ground-truth label: $|r_{t,i} - f_1(\mathbf{x}_{t,i}; \boldsymbol{\theta}_{t-1}^1)|$ and $\text{ReLU}(r_{t,i} - f_1(\mathbf{x}_{t,i}; \boldsymbol{\theta}_{t-1}^1))$. Algorithm 1 depicts the workflow of this EE-Net.

Remark 2 (Network structure) *The structures of the networks f_1 and f_2 can vary depending on the application. For instance, in vision tasks, f_1 can be implemented as transformers (Vaswani et al., 2017). For the exploration network f_2 , the input $\nabla_{\boldsymbol{\theta}^1} f_1$ might have high dimensions when the exploitation network f_1 is wide and deep, leading to substantial computational costs for f_2 . To mitigate this issue, dimensionality reduction techniques can be used to obtain low-dimensional vectors of $\nabla_{\boldsymbol{\theta}^1} f_1$. In our experiments, we employed the locally linear embedding (LLE) method from (Roweis and Saul, 2000) to reduce $\nabla_{\boldsymbol{\theta}^1} f_1$ to a 10-dimensional vector, which achieved the best performance among all baselines. We chose LLE for its ability to preserve local non-linear structures in the high-dimensional gradient space. The resulting embedding vector is denoted by $\phi(\mathbf{x}_{t,i})$ after normalization.*

Remark 3 (Exploration direction) *EE-Net has the capability to determine the direction of exploration. Given an arm $\mathbf{x}_{t,i}$, if the estimate $f_1(\mathbf{x}_{t,i})$ is lower than the expected reward $h(\mathbf{x}_{t,i})$, the learner should increase the likelihood of exploring $\mathbf{x}_{t,i}$ ("upward" exploration). Conversely, if $f_1(\mathbf{x}_{t,i})$ is higher than $h(\mathbf{x}_{t,i})$, the learner should decrease the likelihood of exploring $\mathbf{x}_{t,i}$ ("downward" exploration). EE-Net employs the neural network f_2 to predict $h(\mathbf{x}_{t,i}) - f_1(\mathbf{x}_{t,i})$, which yields positive or negative scores and thus determines the exploration direction.*

Table 1: Structure of EE-Net (Round t).

Input	Network	Label
$\{\mathbf{x}_{\tau,i}\}_{\tau=1}^t$	$f_1(\cdot; \boldsymbol{\theta}^1)$ (Exploitation)	$\{r_\tau\}_{\tau=1}^t$
$\{\nabla_{\boldsymbol{\theta}_{\tau-1}^1} f_1(\mathbf{x}_{\tau,i}; \boldsymbol{\theta}_{\tau-1}^1)\}_{\tau=1}^t$	$f_2(\cdot; \boldsymbol{\theta}^2)$ (Exploration)	$\left\{ \left(r_\tau - f_1(\mathbf{x}_{\tau,i}; \boldsymbol{\theta}_{\tau-1}^1) \right) \right\}_{\tau=1}^t$

Remark 4 (Space complexity) *NeuralUCB and NeuralTS need to maintain the gradient outer product matrix (e.g., $\mathbf{A}_t = \sum_{\tau=1}^t \nabla_{\boldsymbol{\theta}^1} f_1(\mathbf{x}_{\tau,i}; \boldsymbol{\theta}_\tau^1) \nabla_{\boldsymbol{\theta}^1} f_1(\mathbf{x}_{\tau,i}; \boldsymbol{\theta}_\tau^1)^\top \in \mathbb{R}^{p_1 \times p_1}$ and $\boldsymbol{\theta}^1 \in \mathbb{R}^{p_1}$), which has a space complexity of $O(p_1^2)$ to store the outer product. In contrast, EE-Net does not require this matrix and treats $\nabla_{\boldsymbol{\theta}^1} f_1$ only as the input to f_2 . Consequently, EE-Net reduces the space complexity from $O(p_1^2)$ to $O(p_1)$.*

5. Regret Analysis

In this section, we provide the regret analysis of EE-Net. Then, for the analysis, we have the following assumption, which is a standard input assumption in neural bandits and over-parameterized neural networks (Zhou et al., 2020; Allen-Zhu et al., 2019).

Assumption 5.1 *For any $t \in [T], i \in [n], \|\mathbf{x}_{t,i}\|_2 = 1$, and $r_{t,i} \in [0, 1]$.*

The analysis will focus on over-parameterized neural networks (Jacot et al., 2018; Du et al., 2019; Allen-Zhu et al., 2019). Given an input $\mathbf{x} \in \mathbb{R}^d$, we define the fully-connected network f with depth $L \geq 2$ and width m :

$$f(\mathbf{x}; \boldsymbol{\theta}) = \mathbf{W}_L \sigma(\mathbf{W}_{L-1} \sigma(\mathbf{W}_{L-2} \dots \sigma(\mathbf{W}_1 \mathbf{x}))) \quad (5.1)$$

where σ is the ReLU activation function, $\mathbf{W}_1 \in \mathbb{R}^{m \times d}$, $\mathbf{W}_l \in \mathbb{R}^{m \times m}$, for $2 \leq l \leq L-1$, $\mathbf{W}^L \in \mathbb{R}^{1 \times m}$, and $\boldsymbol{\theta} = [\text{vec}(\mathbf{W}_1)^\top, \text{vec}(\mathbf{W}_2)^\top, \dots, \text{vec}(\mathbf{W}_L)^\top]^\top \in \mathbb{R}^p$.

Initialization. For any $l \in [L-1]$, each entry of \mathbf{W}_l is drawn from the normal distribution $\mathcal{N}(0, \frac{2}{m})$ and \mathbf{W}_L is drawn from the normal distribution $\mathcal{N}(0, \frac{1}{m})$. f_1 and f_2 both follow above network structure but with different input and output, denoted by $f_1(\mathbf{x}; \boldsymbol{\theta}^1), \boldsymbol{\theta}^1 \in \mathbb{R}^{p_1}, f_2(\phi(\mathbf{x}); \boldsymbol{\theta}^2), \boldsymbol{\theta}^2 \in \mathbb{R}^{p_2}$. Recall that η_1, η_2 are the learning rates for f_1, f_2 .

Before providing the following theorem, first, we present the definition of function class following (Cao and Gu, 2019), which is closely related to our regret analysis. Given the parameter space of exploration network f_2 , we define the following function class around initialization:

$$\mathcal{B}(\boldsymbol{\theta}_0^2, \omega) = \{\boldsymbol{\theta} \in \mathbb{R}^{p_2} : \|\boldsymbol{\theta} - \boldsymbol{\theta}_0^2\|_2 \leq \omega/m^{1/4}\}.$$

We slightly abuse the notations. Let $(\mathbf{x}_1, r_1), (\mathbf{x}_2, r_2), \dots, (\mathbf{x}_{Tn}, r_{Tn})$ represent all the data in T rounds. Then, we have the following instance-dependent complexity term:

$$\Psi(\boldsymbol{\theta}_0^2, \omega) = \inf_{\boldsymbol{\theta} \in \mathcal{B}(\boldsymbol{\theta}_0^2, \omega)} \sum_{t=1}^{Tn} (f_2(\mathbf{x}_t; \boldsymbol{\theta}) - r_t)^2 \quad (5.2)$$

Then, we provide the following regret bound.

Theorem 5 For any $\delta \in (0, 1)$, $R > 0$, suppose $m \geq \Omega(\text{poly}(T, L, R, n, \log(1/\delta)))$, $\eta_1 = \eta_2 = \frac{R^2}{\sqrt{m}}$. Then, with probability at least $1 - \delta$ over the initialization, the pseudo regret of Algorithm 1 in T rounds satisfies

$$\mathbf{R}_T \leq \sqrt{T} \cdot \mathcal{O}\left(\sqrt{\Psi(\boldsymbol{\theta}_0^2, R)} + \sqrt{2\log(1/\delta)}\right) + \mathcal{O}(1). \quad (5.3)$$

Under the similar assumptions in over-parameterized neural networks, the regret bounds of NeuralUCB (Zhou et al., 2020) and NeuralTS (Zhang et al., 2021) are both

$$\begin{aligned} \mathbf{R}_T &\leq \mathcal{O}\left(\sqrt{\tilde{d}T \log T + S^2}\right) \cdot \mathcal{O}\left(\sqrt{\tilde{d} \log T}\right), \\ S &= \sqrt{2\mathbf{h}\mathbf{H}^{-1}\mathbf{h}} \quad \text{and} \quad \tilde{d} = \frac{\log \det(\mathbf{I} + \mathbf{H}/\lambda)}{\log(1 + Tn/\lambda)} \end{aligned} \quad (5.4)$$

where \tilde{d} is the effective dimension, \mathbf{H} is the neural tangent kernel matrix (NTK, Def.5.1) (Jacot et al., 2018; Arora et al., 2019) formed by the arm contexts of T rounds defined in (Zhou et al., 2020), and λ is a regularization parameter. For the complexity term S , $\mathbf{h} = \{h(\mathbf{x}_i)\}_{i=1}^{Tn}$ and it represent the complexity of data of T rounds. Similarly, in linear contextual bandits, Abbasi-Yadkori et al. (2011) achieves $\mathcal{O}(d\sqrt{T} \log T)$ and Li et al. (2017) achieves $\mathcal{O}(\sqrt{dT} \log T)$.

The complexity term $\Psi(\boldsymbol{\theta}_0^2, R)$ in Theorem 5 is easier to interpret. The complexity term S depends on the smallest eigenvalue of \mathbf{H} and \mathbf{h} and thus it is not straightforward to explain the physical meaning of S . Moreover, as $h(\cdot) \in [0, 1]$, in the worst case, $\|\mathbf{h}\|_2 = \Theta(T)$. Therefore, S^2 can in general grow linearly with T . In contrast, $\Psi(\boldsymbol{\theta}_0^2, R)$ represents the smallest squared regression error a function class achieves, as defined in Eq. 5.2. The parameter R in Theorem 5 shows the size of the function class we can use to achieve small $\Psi(\boldsymbol{\theta}_0^2, R)$. Theorem 5 shows that EE-Net can achieve the optimal $\tilde{\mathcal{O}}(\sqrt{T})$ regret upper bound, if the data can be "well-classified" by the over-parameterized neural network functions, i.e., $\Psi(\boldsymbol{\theta}_0^2, R)$ is a small constant. The instance-dependent terms $\Psi(\boldsymbol{\theta}_0^2, R)$ controlled by R also appears in works (Chen et al., 2021; Cao and Gu, 2019), but their regret bounds directly depend on R , i.e., $\tilde{\mathcal{O}}(\sqrt{T}) + \mathcal{O}(R)$. This indicates that their regret bounds are invalid (change to $\mathcal{O}(T)$) when R has $\mathcal{O}(T)$. On the contrary, we remove the dependence of R in Theorem 5, which enables us to choose R with $\mathcal{O}(T)$ to have broader function classes. This is more realistic, because the parameter space is usually much larger than the number of data points for neural networks (Deng et al., 2009). (Foster and Rakhlin, 2020; Foster and Krishnamurthy, 2021) provide an instance-dependent regret upper bound which requires an online regression oracle, but their analysis is finished in the parametric setting regarding the reward function.

Remark 6 Context assumption. Theorem 5 is notable for not making any assumptions about the contexts $\{\mathbf{x}_t\}_{t=1}^{Tn}$ used in the problem. This lack of restriction allow the arms to be chosen repeatedly. In contrast, existing neural bandit algorithms, such as those in (Zhou et al., 2020; Zhang et al., 2021; Kassraie and Krause, 2022), rely on Assumption 5.2 for the contexts. This assumption requires the Neural Tangent Kernel (NTK) Gram matrix formed by $\{\mathbf{x}_t\}_{t=1}^{Tn}$ to be positive-definite, which means no arm context can be repetitively

observed. Consequently, the regret upper bounds of these algorithms can be easily disrupted by straightforward context attacks, such as creating two identical contexts with different rewards.

Remark 7 Proof workflow. Compared to NeuralUCB/TS, our proof is directly built on recent advances in convergence theory (Allen-Zhu et al., 2019) and generalization bound (Cao and Gu, 2019) of over-parameterized neural networks. Instead, the analysis for NeuralUCB/TS contains three parts of approximation error by calculating the distances among the expected reward, ridge regression, NTK function, and the network function.

The proof of Theorem 5 is in Appendix B.3 and mainly based on the following generalization bound for the exploration neural network. The bound results from an online-to-batch conversion with an instance-dependent complexity term controlled by deep neural networks.

Lemma 8 Suppose m, η_1, η_2 satisfies the conditions in Theorem 5. In round $t \in [T]$, let

$$\hat{i} = \arg \max_{i \in [k]} \left(f_1(\mathbf{x}_{t,\hat{i}}; \boldsymbol{\theta}_{t-1}^1) + f_2(\phi(\mathbf{x}_{t,\hat{i}}); \boldsymbol{\theta}_{t-1}^2) \right),$$

and denote the policy by π_t . Then, for any $\delta \in (0, 1)$, $R > 0$, with probability at least $1 - \delta$, for $t \in [T]$, it holds uniformly

$$\begin{aligned} \frac{1}{t} \sum_{\tau=1}^t \mathbb{E}_{r_{\tau,\hat{i}}} \left[\left| f_1(\mathbf{x}_{\tau,\hat{i}}; \boldsymbol{\theta}_{\tau-1}^1) + f_2(\phi(\mathbf{x}_{\tau,\hat{i}}); \boldsymbol{\theta}_{\tau-1}^2) - r_{\tau,\hat{i}} \right| \mid \pi_t, \mathcal{H}_{\tau-1} \right] \\ \leq \frac{\sqrt{\Psi(\boldsymbol{\theta}_0^2, R)} + \mathcal{O}(1)}{\sqrt{t}} + \sqrt{\frac{2 \log(\mathcal{O}(1)/\delta)}{t}}. \end{aligned} \tag{5.5}$$

where $\mathcal{H}_t = \{\mathbf{x}_{\tau,\hat{i}}, r_{\tau,\hat{i}}\}_{\tau=1}^t$ represents of historical data selected by π_τ and expectation is taken over the reward.

Lemma 8 establishes an instance-dependent generalization bound for exploration networks, exhibiting a complexity term of $\mathcal{O}(t^{-1/2})$. We achieve this by working in the regression rather than the classification setting and utilizing the almost convexity of square loss. Note that the bound in Lemma 8 holds in the setting of bounded (possibly random) rewards $r \in [0, 1]$ instead of a fixed function in the conventional classification setting.

5.1 Connection to Neural Tangent Kernel

In addition, we provide one method to connect $\Psi(\boldsymbol{\theta}_0, R)$ with NTK, or \tilde{d} and S , where $\boldsymbol{\theta}_0$ represents the initialized parameter. The following assumption is widely used in neural contextual bandits (Zhou et al., 2020; Zhang et al., 2021; Kassarai and Krause, 2022). It requires that NTK gram matrix formed by all arm contexts has to be positive-definite, which also implies that no arm context is allowed to be repetitively observed.

Definition 9 (NTK Jacot et al. (2018); Wang et al. (2021b)) Let \mathcal{N} denote the normal distribution. Given the data instances $\{\mathbf{x}_t\}_{t=1}^{Tn}$, for all $i, j \in [Tn]$, define

$$\begin{aligned} \mathbf{H}_{i,j}^0 &= \Sigma_{i,j}^0 = \langle \mathbf{x}_i, \mathbf{x}_j \rangle, \quad \mathbf{A}_{i,j}^l = \begin{pmatrix} \Sigma_{i,i}^l & \Sigma_{i,j}^l \\ \Sigma_{j,i}^l & \Sigma_{j,j}^l \end{pmatrix} \\ \Sigma_{i,j}^l &= 2\mathbb{E}_{a,b \sim \mathcal{N}(\mathbf{0}, \mathbf{A}_{i,j}^{l-1})}[\sigma(a)\sigma(b)], \\ \mathbf{H}_{i,j}^l &= 2\mathbf{H}_{i,j}^{l-1}\mathbb{E}_{a,b \sim \mathcal{N}(\mathbf{0}, \mathbf{A}_{i,j}^{l-1})}[\sigma'(a)\sigma'(b)] + \Sigma_{i,j}^l. \end{aligned}$$

Then, the NTK matrix is defined as $\mathbf{H} = (\mathbf{H}^L + \Sigma^L)/2$.

Assumption 5.2 There exists $\lambda_0 > 0$, such that $\mathbf{H} \succeq \lambda_0 \mathbf{I}$

With this assumption, we can further bound $\Psi(\boldsymbol{\theta}_0, R)$ in Theorem 5 as the following lemma.

Lemma 10 Suppose Assumption 5.2 and conditions in Theorem 5 holds where $m \geq \tilde{\Omega}(\text{poly}(T, L) \cdot n\lambda_0^{-1} \log(1/\delta))$. With probability at least $1 - \delta$ over the initialization, there exists $\boldsymbol{\theta}' \in B(\boldsymbol{\theta}_0, \tilde{\Omega}(T^{3/2}))$, such that

$$\mathbb{E}[\Psi(\boldsymbol{\theta}_0, \tilde{\Omega}(T^{3/2}))] \leq \mathbb{E}\left[\sum_{t=1}^{Tn} (f(\mathbf{x}_t; \boldsymbol{\theta}') - r_t)^2\right] \leq \tilde{\mathcal{O}}\left(\sqrt{\tilde{d}} + S\right)^2 \cdot \tilde{d}.$$

Lemma 10 provides an upper bound for $\Psi(\boldsymbol{\theta}_0, \tilde{\Omega}(T^{3/2}))$ by setting $R \geq \tilde{\Omega}(T^{3/2})$ for a neural network model f . This connection implies that $\mathbb{E}[\mathbf{R}_T] \leq \tilde{\mathcal{O}}(\sqrt{\tilde{d}T}(S + \sqrt{\tilde{d}}))$ in Theorem 5 This also implies that $\mathbb{E}[\mathbf{R}_T]$ in Theorem 5 is at least as good as the upper bounds in (Zhou et al., 2020; Wang et al., 2021b).

6. Experiments

In this section, we evaluate EE-Net on four real-world datasets comparing with strong state-of-the-art baselines. We first present the setup of experiments, then show regret comparison and report ablation study. Codes are publicly available¹.

Baselines. To comprehensively evaluate EE-Net, we choose 4 neural-based bandit algorithms, one linear and one kernelized bandit algorithms.

1. LinUCB (Li et al., 2010) explicitly assumes the reward is a linear function of arm vector and unknown user parameter and then applies the ridge regression and an upper confidence bound to determine selected arm.
2. KernelUCB (Valko et al., 2013) adopts a predefined kernel matrix on the reward space combined with a UCB-based exploration strategy.
3. NeuralNTK is a variant of KernelUCB specialized to the Neural Tangent Kernel.

1. <https://github.com/banyikun/EE-Net-ICLR-2022>

4. Neural-Epsilon adapts the epsilon-greedy exploration strategy on exploitation network f_1 . With probability $1 - \epsilon$, the arm is selected by $\mathbf{x}_t = \arg \max_{i \in [n]} f_1(\mathbf{x}_{t,i}; \boldsymbol{\theta}^1)$ and with probability ϵ , the arm is chosen randomly.
5. NeuralUCB (Zhou et al., 2020) uses the exploitation network f_1 to learn the reward function coming with an UCB-based exploration strategy.
6. NeuralTS (Zhang et al., 2021) adopts the exploitation network f_1 to learn the reward function coming with an Thompson Sampling exploration strategy.

Note that we do not report the results of LinTS and KernelTS in the experiments, because LinTS and KernelTS have been significantly outperformed by NeuralTS (Zhang et al., 2021). **MNIST dataset.** MNIST is a well-known image dataset (LeCun et al., 1998) for the 10-class classification problem. Following the evaluation setting of existing works (Valko et al., 2013; Zhou et al., 2020; Zhang et al., 2021), we transform this classification problem into bandit problem. Consider an image $\mathbf{x} \in \mathbb{R}^d$, we aim to classify it from 10 classes. First, in each round, the image \mathbf{x} is transformed into 10 arms and presented to the learner, represented by 10 vectors in sequence $\mathbf{x}_1 = (\mathbf{x}, \mathbf{0}, \dots, \mathbf{0})$, $\mathbf{x}_2 = (\mathbf{0}, \mathbf{x}, \dots, \mathbf{0})$, \dots , $\mathbf{x}_{10} = (\mathbf{0}, \mathbf{0}, \dots, \mathbf{x}) \in \mathbb{R}^{10d}$. The reward is defined as 1 if the index of selected arm matches the index of \mathbf{x} 's ground-truth class; Otherwise, the reward is 0.

Yelp and MovieLens datasets. Yelp² is a dataset released in the Yelp dataset challenge, which consists of 4.7 million rating entries for 1.57×10^5 restaurants by 1.18 million users. MovieLens(Harper and Konstan, 2015) is a dataset consisting of 25 million ratings between 1.6×10^5 users and 6×10^4 movies. We build the rating matrix by choosing the top 2000 users and top 10000 restaurants(movies) and use singular-value decomposition (SVD) to extract a 10-dimension feature vector for each user and restaurant(movie). In these two datasets, the bandit algorithm is to choose the restaurants(movies) with bad ratings. This is similar to the recommendation of good restaurants by catching the bad restaurants. We generate the reward by using the restaurant(movie)'s gained stars scored by the users. In each rating record, if the user scores a restaurant(movie) less than 2 stars (5 stars totally), its reward is 1; Otherwise, its reward is 0. In each round, we set 10 arms as follows: we randomly choose one with reward 1 and randomly pick the other 9 restaurants(movies) with 0 rewards; then, the representation of each arm is the concatenation of the corresponding user feature vector and restaurant(movie) feature vector.

Disin dataset. Disin(Ahmed et al., 2018) is a fake news dataset on kaggle³ including 12600 fake news articles and 12600 truthful news articles, where each article is represented by the text. To transform the text into vectors, we use the approach (Fu and He, 2021) to represent each article by a 300-dimension vector. Similarly, we form a 10-arm pool in each round, where 9 real news and 1 fake news are randomly selected. If the fake news is selected, the reward is 1; Otherwise, the reward is 0.

Configurations. For LinUCB, following (Li et al., 2010), we do a grid search for the exploration constant α over (0.01, 0.1, 1) which is to tune the scale of UCB. For KernelUCB (Valko et al., 2013), we use the radial basis function kernel and stop adding contexts after 1000 rounds, following (Valko et al., 2013; Zhou et al., 2020). For the regularization parameter

2. <https://www.yelp.com/dataset>

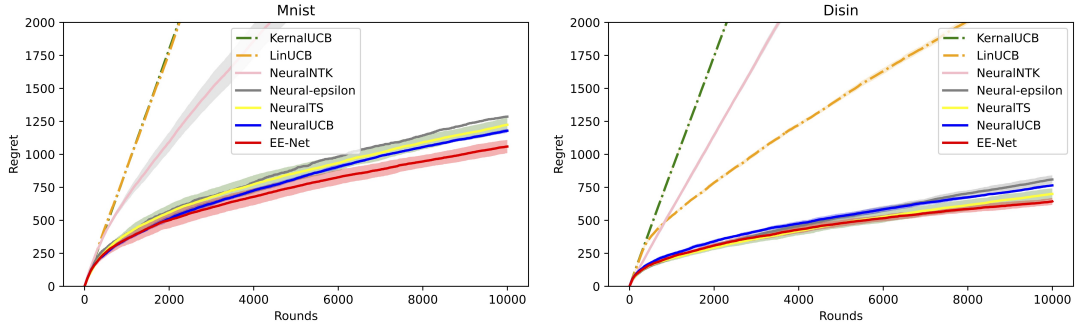
3. <https://www.kaggle.com/clmentbisailon/fake-and-real-news-dataset>

Table 2: Cumulative regret of all methods in 10000 rounds.

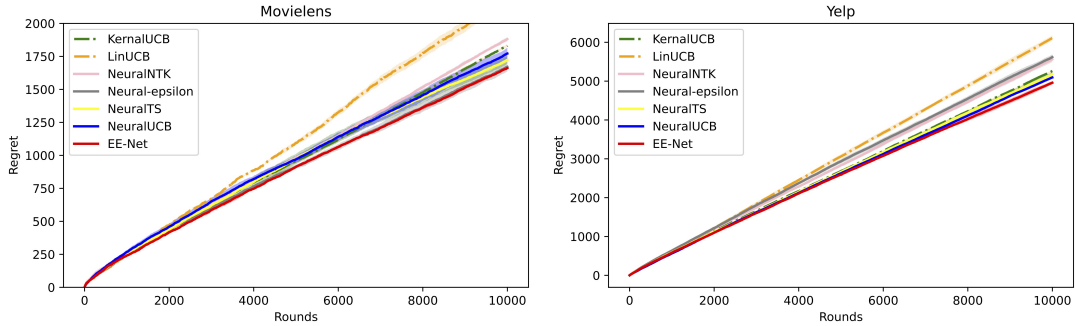
	MNIST	Disin	Movielens	Yelp
LinUCB	7863.2 \pm 32	2457.9 \pm 11	2143.5 \pm 35	5917.0 \pm 34
KenelUCB	7635.3 \pm 28	8219.7 \pm 21	1723.4 \pm 3	4872.3 \pm 11
NeuralNTK	4692.3 \pm 78	6736.7 \pm 49	1879.4 \pm 13	5253.2 \pm 41
Neural- ϵ	1126.8 \pm 6	734.2 \pm 31	1573.4 \pm 26	5276.1 \pm 27
NeuralUCB	943.5 \pm 8	641.7 \pm 23	1654.0 \pm 31	4593.1 \pm 13
NeuralTS	965.8 \pm 87	523.2 \pm 43	1583.1 \pm 23	4676.6 \pm 7
EE-Net	842.3 \pm 72(10.7% \uparrow)	476.4 \pm 23(8.9% \uparrow)	1472.4 \pm 5(6.4% \uparrow)	4403.1 \pm 13(4.1% \uparrow)

λ and exploration parameter ν in KernelUCB, we do the grid search for λ over (0.1, 1, 10) and for ν over (0.01, 0.1, 1). For NeuralUCB and NeuralTS, following setting of (Zhou et al., 2020; Zhang et al., 2021), we use the exploitation network f_1 and conduct the grid search for the exploration parameter ν over (0.001, 0.01, 0.1, 1) and for the regularization parameter λ over (0.01, 0.1, 1). For Neural- ϵ , we use the same neural network f_1 and do the grid search for the exploration probability ϵ over (0.01, 0.1, 0.2). For the neural bandits NeuralUCB/TS, following their setting, as they have expensive computation cost to store and compute the whole gradient matrix, we use a diagonal matrix to make approximation. For all neural networks, we conduct the grid search for learning rate over (0.01, 0.001, 0.0005, 0.0001). For all grid-searched parameters, we choose the best of them for the comparison and report the averaged results of 10 runs for all methods. To compare fairly, for all the neural-based methods, including EE-Net, the exploitation network f_1 is built by a 2-layer fully-connected network with 100 width. For the exploration network f_2 , we use a 2-layer fully-connected network with 100 width as well. In the training process, we update neural networks according to all past observed data, following (Zhou et al., 2020).

Results. Figures 2 and 3 present the average cumulative regrets of all methods over 10,000 rounds, while Figures 2a and 2b compare regrets across four datasets. Table 2 reports the final regret of all methods. EE-Net consistently outperforms all baselines. LinUCB and KernelUCB struggle due to their reliance on a simple linear reward function or predefined kernel, which fail to capture the complex reward structures in real-world datasets. This limitation is particularly evident in the MNIST and Disin datasets, where reward-arm correlations are neither linear nor simple mappings. As a result, these methods fail to effectively leverage past data and select optimal arms. Among neural-based bandit algorithms, NeuralNTK performs poorly in practice due to its reliance on over-parameterized assumptions, which are often impractical. Neural- ϵ relies on random exploration, failing to utilize available state information. NeuralUCB and NeuralTS attempt to address exploration through statistical confidence bounds. NeuralUCB computes a gradient-based upper confidence bound, while NeuralTS samples predicted rewards from a normal distribution with a gradient-based standard deviation. However, these approaches primarily account for worst-case deviations and may not adaptively estimate each arm’s potential gain. In contrast, EE-Net leverages a neural network f_2 to learn arm potentials through powerful representation learning, enabling adaptive exploration. This allows EE-Net to outperform state-of-the-art bandit algorithms. Additionally, while NeuralUCB and NeuralTS require tuning two parameters for confidence



(a) Regret comparison on MNIST and Disin.



(b) Regret comparison on Movielens and Yelp.

Figure 2: With the same exploitation network f_1 , EE-Net outperforms neural-based baselines.

bounds or standard deviations across different tasks, EE-Net simply sets up a neural network that autonomously learns optimal exploration strategies.

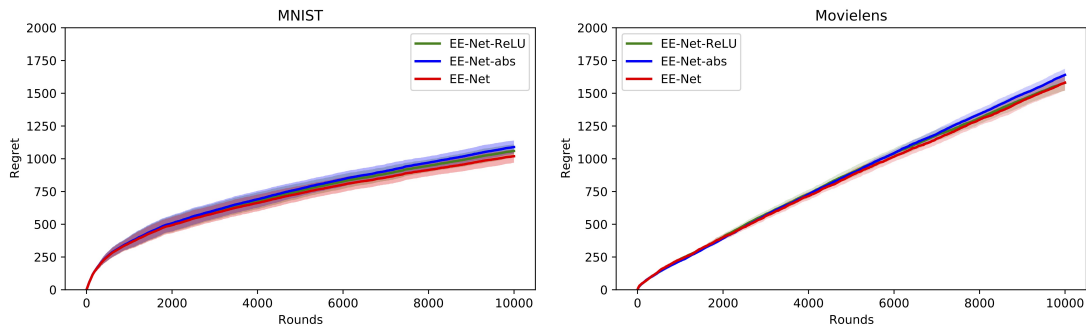


Figure 3: Ablation study on label function y for f_2 . EE-Net denotes $y_1 = r - f_1$, EE-Net-abs denotes $y_2 = |r - f_1|$, and EE-Net-ReLU denotes $y_3 = \text{ReLU}(r - f_1)$. EE-Net shows the best performance on these two datasets.

Ablation study for y . In this paper, we use $y_1 = r - f_1$ to measure the potential gain of an arm, as the label of f_2 . Moreover, we provide other two intuitive form $y_2 = |r - f_1|$ and $y_3 = \text{ReLU}(r - f_1)$. Figure 3 shows the regret with different y , where "EE-Net" denotes our method with default y_1 , "EE-Net-abs" represents the one with y_2 and "EE-Net-ReLU" is with y_3 . On Movielens and MNIST datasets, EE-Net slightly outperforms EE-Net-abs and

EE-Net-ReLU. In fact, y_1 can effectively represent the positive potential gain and negative potential gain, such that f_2 intends to score the arm with positive gain higher and score the arm with negative gain lower. However, y_2 treats the positive/negative potential gain evenly, weakening the discriminative ability. y_3 can recognize the positive gain while neglecting the difference of negative gain. Therefore, y_1 usually is the most effective one for empirical performance.

Table 3: Different dimensionality of input of exploration network

Dimensionality	10	50	200	500
Regret	1472.3 ± 5	1463.5 ± 6.3	1452.1 ± 4.2	1467.6 ± 2.4

Dimension reduction. We conducted additional experiments with varying levels of dimension reduction for $\nabla_{\theta^1} f_1$. A grid search was performed over the set $\{10, 20, 100, 500\}$ using the Movielens dataset. Table 3 illustrates the performance changes with increasing dimensionality. The results indicate that higher dimensionality (from 10 to 200) can enhance performance because the feature embedding captures more information. However, this comes at the cost of increased computational requirements. As the complexity of the exploitation network f_1 increases, a higher input dimensionality becomes necessary. However, a higher-dimensional input may demand a more complex architecture for f_2 to effectively learn the intricate mapping. This could explain the decline in performance observed when the dimensionality reaches 500. Therefore, a balance should be made between performance and computation complexity, depending on the specific models and applications.

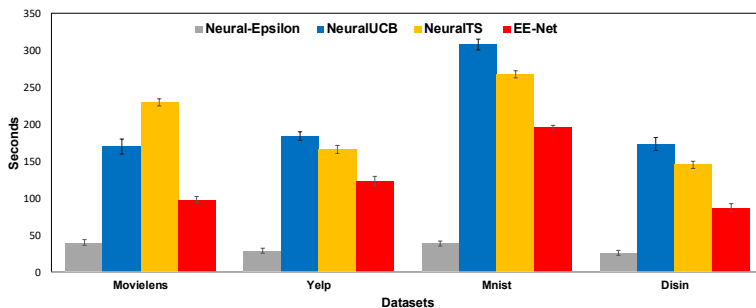


Figure 4: Decision-making time

Running time analysis. During training, EE-Net incurs a higher cost since it requires training an additional neural network (the Exploration network), leading to approximately 38–46% more computation compared to NeuralUCB and NeuralTS. In contrast, during inference, EE-Net is more efficient. NeuralUCB and NeuralTS must compute the inverse of the gradient outer product matrix at each decision step, whereas EE-Net only performs a single forward pass of the Exploration network. As a result, As shown in Figure 4, EE-Net achieves 32–59% faster inference compared to NeuralUCB and NeuralTS. This efficiency advantage is particularly important in real-world applications that demand fast decision-making and responsiveness.

7. Conclusion

In this paper, we propose a novel exploration strategy, EE-Net, by investigating the exploration direction in contextual bandits. In addition to a neural network that exploits collected data in past rounds, EE-Net has another neural network to learn the potential gain compared to the current estimate for adaptive exploration. We provide an instance-dependent regret upper bound for EE-Net and then use experiments to demonstrate its empirical performance.

Acknowledgements

We are grateful to Shiliang Zuo and Yunzhe Qi for the valuable discussions in the revisions of manuscript. This research work is supported by National Science Foundation under Award No. IIS-1947203, IIS-2002540, IIS-2137468, IIS-1908104, OAC-1934634, DBI 20-21898, and GEO 2530555, and a grant from C3.ai. The views and conclusions are those of the authors and should not be interpreted as representing the official policies of the funding agencies or the government.

References

- Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. In *Advances in Neural Information Processing Systems*, pages 2312–2320, 2011.
- Marc Abeille and Alessandro Lazaric. Linear thompson sampling revisited. In *Artificial Intelligence and Statistics*, pages 176–184. PMLR, 2017.
- Shipra Agrawal and Navin Goyal. Thompson sampling for contextual bandits with linear payoffs. In *International Conference on Machine Learning*, pages 127–135. PMLR, 2013.
- Hadeer Ahmed, Issa Traore, and Sherif Saad. Detecting opinion spams and fake news using text classification. *Security and Privacy*, 1(1):e9, 2018.
- Zeyuan Allen-Zhu, Yuanzhi Li, and Zhao Song. A convergence theory for deep learning via over-parameterization. In *International Conference on Machine Learning*, pages 242–252. PMLR, 2019.
- Sanjeev Arora, Simon S Du, Wei Hu, Zhiyuan Li, Russ R Salakhutdinov, and Ruosong Wang. On exact computation with an infinitely wide neural net. In *Advances in Neural Information Processing Systems*, pages 8141–8150, 2019.
- Peter Auer. Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research*, 3(Nov):397–422, 2002.
- Yikun Ban and Jingrui He. Generic outlier detection in multi-armed bandit. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 913–923, 2020.
- Yikun Ban and Jingrui He. Convolutional neural bandit: Provable algorithm for visual-aware advertising. *arXiv preprint arXiv:2107.07438*, 2021a.

- Yikun Ban and Jingrui He. Local clustering in contextual multi-armed bandits. In *Proceedings of the Web Conference 2021*, pages 2335–2346, 2021b.
- Yikun Ban, Jingrui He, and Curtiss B Cook. Multi-facet contextual bandits: A neural network perspective. In *The 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, Singapore, August 14-18, 2021*, pages 35–45, 2021.
- Yikun Ban, Yuchen Yan, Arindam Banerjee, and Jingrui He. EE-net: Exploitation-exploration neural networks in contextual bandits. In *International Conference on Learning Representations*, 2022a. URL https://openreview.net/forum?id=X_ch3VrNSRg.
- Yikun Ban, Yuheng Zhang, Hanghang Tong, Arindam Banerjee, and Jingrui He. Improved algorithms for neural active learning. *Advances in Neural Information Processing Systems*, 35:27497–27509, 2022b.
- Yikun Ban, Ishika Agarwal, Ziwei Wu, Yada Zhu, Kommy Weldemariam, Hanghang Tong, and Jingrui He. Neural active learning beyond bandits. *arXiv preprint arXiv:2404.12522*, 2024a.
- Yikun Ban, Yunzhe Qi, and Jingrui He. Neural contextual bandits for personalized recommendation. In *Companion Proceedings of the ACM on Web Conference 2024*, pages 1246–1249, 2024b.
- Yikun Ban, Yunzhe Qi, Tianxin Wei, Lihui Liu, and Jingrui He. Meta clustering of neural bandits. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 95–106, 2024c.
- Yikun Ban, Jiaru Zou, Zihao Li, Yunzhe Qi, Dongqi Fu, Jian Kang, Hanghang Tong, and Jingrui He. Pagerank bandits for link prediction. *Advances in Neural Information Processing Systems*, 37:21342–21376, 2024d.
- Djallel Bouneffouf, Irina Rish, and Charu Aggarwal. Survey on applications of multi-armed and contextual bandits. In *2020 IEEE Congress on Evolutionary Computation (CEC)*, pages 1–8. IEEE, 2020.
- Sébastien Bubeck, Rémi Munos, Gilles Stoltz, and Csaba Szepesvári. X-armed bandits. *Journal of Machine Learning Research*, 12(5), 2011.
- Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. Exploration by random network distillation. *arXiv preprint arXiv:1810.12894*, 2018.
- Yuan Cao and Quanquan Gu. Generalization bounds of stochastic gradient descent for wide and deep neural networks. *Advances in Neural Information Processing Systems*, 32: 10836–10846, 2019.
- Xinyi Chen, Edgar Minasyan, Jason D Lee, and Elad Hazan. Provable regret bounds for deep online learning and control. *arXiv preprint arXiv:2110.07807*, 2021.
- Wei Chu, Lihong Li, Lev Reyzin, and Robert Schapire. Contextual bandits with linear payoff functions. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pages 208–214, 2011.

- Zhongxiang Dai, Yao Shu, Arun Verma, Flint Xiaofeng Fan, Bryan Kian Hsiang Low, and Patrick Jaillet. Federated neural bandit. *arXiv preprint arXiv:2205.14309*, 2022.
- Rohan Deb, Yikun Ban, Shiliang Zuo, Jingrui He, and Arindam Banerjee. Contextual bandits with online neural regression. *arXiv preprint arXiv:2312.07145*, 2023.
- Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.
- Simon Du, Jason Lee, Haochuan Li, Liwei Wang, and Xiyu Zhai. Gradient descent finds global minima of deep neural networks. In *International Conference on Machine Learning*, pages 1675–1685. PMLR, 2019.
- Vikranth Dwaracherla, Zheng Wen, Ian Osband, Xiuyuan Lu, Seyed Mohammad Asghari, and Benjamin Van Roy. Ensembles for uncertainty estimation: Benefits of prior functions and bootstrapping. *arXiv preprint arXiv:2206.03633*, 2022.
- Vikranth Dwaracherla, Seyed Mohammad Asghari, Botao Hao, and Benjamin Van Roy. Efficient exploration for llms. *arXiv preprint arXiv:2402.00396*, 2024.
- Sarah Filippi, Olivier Cappé, Aurélien Garivier, and Csaba Szepesvári. Parametric bandits: The generalized linear case. In *Advances in Neural Information Processing Systems*, pages 586–594, 2010.
- Dylan Foster and Alexander Rakhlin. Beyond ucb: Optimal and efficient contextual bandits with regression oracles. In *International Conference on Machine Learning*, pages 3199–3210. PMLR, 2020.
- Dylan J Foster and Akshay Krishnamurthy. Efficient first-order contextual bandits: Prediction, allocation, and triangular discrimination. *Advances in Neural Information Processing Systems*, 34, 2021.
- Dongqi Fu and Jingrui He. SDG: A simplified and dynamic graph neural network. In *SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021*, pages 2273–2277. ACM, 2021.
- Quanquan Gu, Amin Karbasi, Khashayar Khosravi, Vahab Mirrokni, and Dongruo Zhou. Batched neural bandits. *ACM/IMS Journal of Data Science*, 1(1):1–18, 2024.
- F Maxwell Harper and Joseph A Konstan. The movielens datasets: History and context. *Acm transactions on interactive intelligent systems (tiis)*, 5(4):1–19, 2015.
- Zixuan Huang, Yikun Ban, Lean Fu, Xiaojie Li, Zhongxiang Dai, Jianxin Li, and deqing wang. Adaptive batch-wise sample scheduling for direct preference optimization. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL <https://openreview.net/forum?id=8FN25P1ktS>.

- Taehyun Hwang, Kyuwook Chai, and Min-hwan Oh. Combinatorial neural bandits. In *International Conference on Machine Learning*, pages 14203–14236. PMLR, 2023.
- Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. In *Advances in neural information processing systems*, pages 8571–8580, 2018.
- Yiling Jia, Weitong ZHANG, Dongruo Zhou, Quanquan Gu, and Hongning Wang. Learning neural contextual bandits through perturbed rewards. In *International Conference on Learning Representations*, 2021.
- Marc Jourdan, Rémy Degenne, Dorian Baudry, Rianne de Heide, and Emilie Kaufmann. Top two algorithms revisited. *Advances in Neural Information Processing Systems*, 35: 26791–26803, 2022.
- Parnian Kassraie and Andreas Krause. Neural contextual bandits without regret. In *International Conference on Artificial Intelligence and Statistics*, pages 240–278. PMLR, 2022.
- Parnian Kassraie, Andreas Krause, and Ilija Bogunovic. Graph neural network bandits. *arXiv preprint arXiv:2207.06456*, 2022.
- Emilie Kaufmann, Olivier Cappé, and Aurélien Garivier. On bayesian upper confidence bounds for bandit problems. In *Artificial intelligence and statistics*, pages 592–600. PMLR, 2012.
- Branislav Kveton, Csaba Szepesvari, Sharan Vaswani, Zheng Wen, Tor Lattimore, and Mohammad Ghavamzadeh. Garbage in, reward out: Bootstrapping exploration in multi-armed bandits. In *International Conference on Machine Learning*, pages 3601–3610. PMLR, 2019.
- Branislav Kveton, Mikhail Konobeev, Manzil Zaheer, Chih-wei Hsu, Martin Mladenov, Craig Boutilier, and Csaba Szepesvari. Meta-thompson sampling. In *International Conference on Machine Learning*, pages 5884–5893. PMLR, 2021.
- John Langford and Tong Zhang. The epoch-greedy algorithm for multi-armed bandits with side information. In *Advances in neural information processing systems*, pages 817–824, 2008.
- Tor Lattimore and Csaba Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020.
- Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.
- Lihong Li, Wei Chu, John Langford, and Robert E Schapire. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international conference on World wide web*, pages 661–670, 2010.
- Lihong Li, Yu Lu, and Dengyong Zhou. Provably optimal algorithms for generalized linear contextual bandits. In *International Conference on Machine Learning*, pages 2071–2080. PMLR, 2017.

- Shuai Li, Alexandros Karatzoglou, and Claudio Gentile. Collaborative filtering bandits. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, pages 539–548, 2016.
- Xiaoqiang Lin, Zhaoxuan Wu, Zhongxiang Dai, Wenyang Hu, Yao Shu, See-Kiong Ng, Patrick Jaillet, and Bryan Kian Hsiang Low. Use your instinct: Instruction optimization using neural bandits coupled with transformers. *arXiv preprint arXiv:2310.02905*, 2023.
- Xiaodong Lu, Xiaohan Wang, Jiajun Chai, Guojun Yin, Wei Lin, Zhijun Chen, Yu Luo, Fuzhen Zhuang, Yikun Ban, and Deqing Wang. Contextual rollout bandits for reinforcement learning with verifiable rewards. *arXiv preprint arXiv:2602.08499*, 2026.
- Xiuyuan Lu and Benjamin Van Roy. Ensemble sampling. *arXiv preprint arXiv:1705.07347*, 2017.
- Ian Osband, John Aslanides, and Albin Cassirer. Randomized prior functions for deep reinforcement learning. *Advances in Neural Information Processing Systems*, 31, 2018.
- Ian Osband, Zheng Wen, Seyed Mohammad Asghari, Vikranth Dwaracherla, Morteza Ibrahimi, Xiuyuan Lu, and Benjamin Van Roy. Approximate thompson sampling via epistemic neural networks. In *Uncertainty in Artificial Intelligence*, pages 1586–1595. PMLR, 2023a.
- Ian Osband, Zheng Wen, Seyed Mohammad Asghari, Vikranth Dwaracherla, Morteza Ibrahimi, Xiuyuan Lu, and Benjamin Van Roy. Epistemic neural networks. *Advances in Neural Information Processing Systems*, 36:2795–2823, 2023b.
- Yunzhe Qi, Yikun Ban, and Jingrui He. Neural bandit with arm group graph. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 1379–1389, 2022.
- Yunzhe Qi, Yikun Ban, and Jingrui He. Graph neural bandits. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '23, page 1920–1931, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9798400701030. doi: 10.1145/3580305.3599371. URL <https://doi.org/10.1145/3580305.3599371>.
- Yunzhe Qi, Yikun Ban, Arindam Banerjee, and Jingrui He. Robust neural contextual bandit against adversarial corruptions. *Advances in Neural Information Processing Systems*, 37: 19378–19446, 2024a.
- Yunzhe Qi, Yikun Ban, Tianxin Wei, Jiaru Zou, Huaxiu Yao, and Jingrui He. Meta-learning with neural bandit scheduler. *Advances in Neural Information Processing Systems*, 36, 2024b.
- Yunzhe Qi, Yao Zhou, Yikun Ban, Allan Stewart, Chuanwei Ruan, Jiachuan He, Shishir Kumar Prasad, Haixun Wang, and Jingrui He. Bi-level hierarchical neural contextual bandits for online recommendation. *Transactions on Machine Learning Research*, 2026. ISSN 2835-8856. URL <https://openreview.net/forum?id=k3XsA75SGv>. J2C Certification.

- Carlos Riquelme, George Tucker, and Jasper Snoek. Deep bayesian bandits showdown: An empirical comparison of bayesian deep networks for thompson sampling. *arXiv preprint arXiv:1802.09127*, 2018.
- Sam T Roweis and Lawrence K Saul. Nonlinear dimensionality reduction by locally linear embedding. *science*, 290(5500):2323–2326, 2000.
- Daniel Russo. Simple bayesian algorithms for best arm identification. In *Conference on learning theory*, pages 1417–1418. PMLR, 2016.
- Aleksandrs Slivkins et al. Introduction to multi-armed bandits. *Foundations and Trends® in Machine Learning*, 12(1-2):1–286, 2019.
- William R Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3/4):285–294, 1933.
- Michal Valko, Nathaniel Korda, Rémi Munos, Ilias Flaounas, and Nelo Cristianini. Finite-time analysis of kernelised contextual bandits. *arXiv preprint arXiv:1309.6869*, 2013.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.
- Zhilei Wang, Pranjal Awasthi, Christoph Dann, Ayush Sekhari, and Claudio Gentile. Neural active learning with performance guarantees. *Advances in Neural Information Processing Systems*, 34:7510–7521, 2021a.
- Zhilei Wang, Pranjal Awasthi, Christoph Dann, Ayush Sekhari, and Claudio Gentile. Neural active learning with performance guarantees. *Advances in Neural Information Processing Systems*, 34, 2021b.
- Qingyun Wu, Huazheng Wang, Quanquan Gu, and Hongning Wang. Contextual bandits in a collaborative environment. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, pages 529–538, 2016.
- Pan Xu, Zheng Wen, Handong Zhao, and Quanquan Gu. Neural contextual bandits with deep representation and shallow exploration. *arXiv preprint arXiv:2012.01780*, 2020.
- Fengkai Yang, Zherui Chen, Xiaohan Wang, Xiaodong Lu, Jiajun Chai, Guojun Yin, Wei Lin, Shuai Ma, Fuzhen Zhuang, Deqing Wang, et al. Your group-relative advantage is biased. *arXiv preprint arXiv:2601.08521*, 2026.
- Weitong Zhang, Dongruo Zhou, Lihong Li, and Quanquan Gu. Neural thompson sampling. In *International Conference on Learning Representations*, 2021.
- Dongruo Zhou, Lihong Li, and Quanquan Gu. Neural contextual bandits with ucb-based exploration. In *International Conference on Machine Learning*, pages 11492–11502. PMLR, 2020.

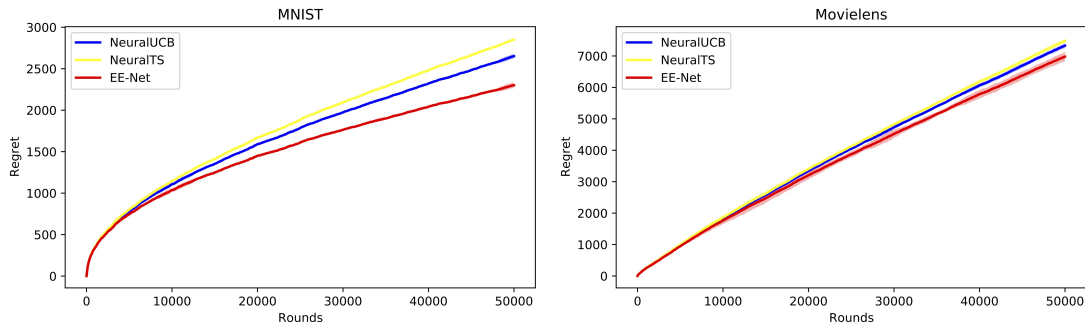


Figure 5: Extended rounds on Movielens and MNIST Datasets

Appendix A. Further Discussion

Figure 5 illustrates the extended rounds of regret compression on the Movielens and MNIST datasets, supporting our claim that the regret upper bound for EE-Net is tighter than those for NeuralUCB and NeuralTS. As depicted in the figure, EE-Net achieves the fastest convergence rate.

A.1 Difference from Existing Works

As the linear bandits (Gentile et al., 2014; Li et al., 2016; Gentile et al., 2017; Li et al., 2019;) work in the Euclidean space and build the confidence ellipsoid for θ^* (optimal parameters) based on the linear function $\mathbb{E}[r_{t,i} | \mathbf{x}_{t,i}] = \langle \mathbf{x}_{t,i}, \theta^* \rangle$, their regret bounds contain d because $\mathbf{x}_{t,i} \in \mathbb{R}^d$. Similarly, neural bandits (Zhou et al., 2020; Zhang et al., 2021) work in the RKHS and construct the confidence ellipsoid for θ^* (neural parameters) according to the linear function $\mathbb{E}[r_{t,i} | \mathbf{x}_{t,i}] = \langle \nabla_{\theta_0} f(\mathbf{x}_{t,i}; \theta_0), \theta^* - \theta_0 \rangle$, where $\nabla_{\theta_0} f(\mathbf{x}_{t,i}; \theta_0) \in \mathbb{R}^p$. Their analysis is built on the NTK approximation, which is a linear approximation with respect to the gradient. Thus their regret bounds are affected by \tilde{d} due to $\nabla_{\theta_0} f(\mathbf{x}_{t,i}; \theta_0) \in \mathbb{R}^p$. On the contrary, our analysis is based on the convergence error (regression) and generalization bound of the neural networks. The convergence error term is controlled by the complexity term Ψ^* . And the generalization bound is the standard large-deviation bound, which depends on the number of data points (rounds).

A.2 Upward and Downward Exploration

Table 4

Datasets	Upward Exploration	Downward Exploration
Mnist	76.3 %	23.7 %
Disin	29.1 %	70.9 %
MovieLens	58.6 %	41.4 %
Yelp	55.3 %	44.7 %

Table 4 shows the proportions of upward and downward explorations recorded over 10,000 rounds in real-world datasets. In each round, we compared the reward estimated by the

exploitation model f_1 with the received reward for each arm, determining the exploration direction. The varying proportions across datasets suggest that identifying the exploration direction can provide more information and may be beneficial for balancing exploitation and exploration, beyond just considering exploration strength.

A.3 Evaluation of Neural-Epsilon

Table 5: Cumulative regret of Neural- ϵ variant.

	MNIST	Disin	Movielens	Yelp
Neural- ϵ	1126.8 \pm 6	734.2 \pm 31	1573.4 \pm 26	5276.1 \pm 27
Neural- ϵ^+	1112.8 \pm 8	724.2 \pm 24	1578.4 \pm 19	5162.4 \pm 43
EE-Net	842.3 \pm 72	476.4 \pm 23	1472.4 \pm 5	4403.1 \pm 13

We also evaluate Neural- ϵ , where ϵ is a decaying function of t . The key intuition behind this approach is that exploration diminishes as more arms and rewards are observed. Specifically, we define ϵ as $\epsilon = \frac{\epsilon_0}{1+\sqrt{t}}$, where ϵ_0 is a constant, and perform a grid search over $\epsilon_0 \in \{0.01, 0.1, 0.2\}$. While this trade-off between exploration and exploitation allows Neural- ϵ^+ to achieve slightly better performance than Neural- ϵ , as shown in Table 5, it still falls short of EE-Net due to the unchanged random exploration mechanism.

A.4 Motivation of Exploration Network

Table 6: Selection Criterion Comparison (\mathbf{x}_t : selected arm in round t).

Methods	Selection Criterion
Neural- ϵ	With probability $1 - \epsilon$, $\mathbf{x}_t = \arg \max_{\mathbf{x}_{t,i}, i \in [n]} f_1(\mathbf{x}_{t,i}; \boldsymbol{\theta}^1)$; Otherwise, select \mathbf{x}_t randomly.
NeuralTS (Zhang et al., 2021)	For $\mathbf{x}_{t,i}, \forall i \in [n]$, draw $\hat{r}_{t,i}$ from $\mathcal{N}(f_1(\mathbf{x}_{t,i}; \boldsymbol{\theta}^1), \sigma_{t,i}^2)$. Then, select $\mathbf{x}_{t,\hat{i}}, \hat{i} = \arg \max_{i \in [n]} \hat{r}_{t,i}$.
NeuralUCB (Zhou et al., 2020)	$\mathbf{x}_t = \arg \max_{\mathbf{x}_{t,i}, i \in [n]} \left(f_1(\mathbf{x}_{t,i}; \boldsymbol{\theta}^1) + \gamma_1 \ \nabla_{\boldsymbol{\theta}^1} f(\mathbf{x}_{t,i}; \boldsymbol{\theta}_t)\ _{\mathbf{A}_t^{-1}} \right)$.
EE-Net	$\mathbf{x}_t = \arg \max_{\mathbf{x}_{t,i}, i \in [n]} \left(f_1(\mathbf{x}_{t,i}; \boldsymbol{\theta}^1) + f_2(\nabla_{\boldsymbol{\theta}^1} f_1(\mathbf{x}_{t,i}; \boldsymbol{\theta}^1); \boldsymbol{\theta}^2) \right)$.

In this section, we list one gradient-based UCB from existing works (Ban et al., 2021; Zhou et al., 2020), which motivates our design of exploration network f_2 .

Lemma 11 (Lemma 5.2 in (Ban et al., 2021)). *Given a set of context vectors $\{\mathbf{x}_t\}_{t=1}^T$ and the corresponding rewards $\{r_t\}_{t=1}^T$, $\mathbb{E}(r_t) = h(\mathbf{x}_t)$ for any $\mathbf{x}_t \in \{\mathbf{x}_t\}_{t=1}^T$. Let $f(\mathbf{x}_t; \boldsymbol{\theta})$ be the L -layers fully-connected neural network where the width is m , the learning rate is η , the number of iterations of gradient descent is K . Then, there exist positive constants C_1, C_2, S , such that if*

$$m \geq \text{poly}(T, n, L, \log(1/\delta)) \cdot d \cdot e^{\sqrt{\log 1/\delta}}, \quad \eta = \mathcal{O}(TmL + m\lambda)^{-1}, \quad K \geq \tilde{\mathcal{O}}(TL/\lambda),$$

then, with probability at least $1 - \delta$, for any $\mathbf{x}_{t,i}$, we have the following upper confidence bound:

$$|h(\mathbf{x}_{t,i}) - f(\mathbf{x}_{t,i}; \boldsymbol{\theta}_t)| \leq \gamma_1 \|\nabla_{\boldsymbol{\theta}_t} f(\mathbf{x}_{t,i}; \boldsymbol{\theta}_t) / \sqrt{m}\|_{\mathbf{A}_t^{-1}} + \gamma_2 + \gamma_1 \gamma_3 + \gamma_4, \quad (\text{A.1})$$

where

$$\gamma_1(m, L) = (\lambda + t\mathcal{O}(L)) \cdot ((1 - \eta m \lambda)^{J/2} \sqrt{t/\lambda}) + 1$$

$$\gamma_2(m, L, \delta) = \|\nabla_{\boldsymbol{\theta}_t} f(\mathbf{x}_{t,i}; \boldsymbol{\theta}_t) / \sqrt{m}\|_{\mathbf{A}_t^{-1}} \cdot \left(\sqrt{\log \left(\frac{\det(\mathbf{A}'_t)}{\det(\lambda \mathbf{I})} \right)} - 2 \log \delta + \lambda^{1/2} S \right)$$

$$\gamma_3(m, L) = C_2 m^{-1/6} \sqrt{\log m} t^{1/6} \lambda^{-7/6} L^{7/2}, \quad \gamma_4(m, L) = C_1 m^{-1/6} \sqrt{\log m} t^{2/3} \lambda^{-2/3} L^3$$

$$\mathbf{A}_t = \lambda \mathbf{I} + \sum_{i=1}^t \nabla_{\boldsymbol{\theta}_t} f(\mathbf{x}_{t,i}; \boldsymbol{\theta}_t) \nabla_{\boldsymbol{\theta}_t} f(\mathbf{x}_{t,i}; \boldsymbol{\theta}_t)^\top / m, \quad \mathbf{A}'_t = \lambda \mathbf{I} + \sum_{i=1}^t \nabla_{\boldsymbol{\theta}_0} f(\mathbf{x}_{t,i}; \boldsymbol{\theta}_0) \nabla_{\boldsymbol{\theta}_0} f(\mathbf{x}_{t,i}; \boldsymbol{\theta}_0)^\top / m.$$

Note that $\nabla_{\boldsymbol{\theta}_0} f(\mathbf{x}_{t,i}; \boldsymbol{\theta}_0)$ is the gradient at initialization. Therefore, the above UCB can be represented as the following form for exploitation network f_1 : $|h(\mathbf{x}_{t,i}) - f_1(\mathbf{x}_{t,i}; \boldsymbol{\theta}_t^1)| \leq \kappa(\nabla_{\boldsymbol{\theta}_t^1} f_1(\mathbf{x}_{t,i}; \boldsymbol{\theta}_t^1))$, where κ is a mapping function.

Given an arm x , let $f_1(x)$ be the estimated reward and $h(x)$ be the expected reward. The exploration network f_2 in EE-Net is to learn $h(x) - f_1(x)$, i.e., the residual between expected reward and estimated reward, which is the ultimate goal of making exploration. There are advantages of using a network f_2 to learn $h(x) - f_1(x)$ in EE-Net, compared to giving a statistical upper bound for it such as NeuralUCB, (Ban et al., 2021), and NeuralTS (in NeuralTS, the variance ν can be thought of as the upper bound). For EE-Net, the approximation error for $h(x) - f_1(x)$ is caused by the generalization error of the neural network (Lemma B.1. in the manuscript). In contrast, for NeuralUCB, (Ban et al., 2021), and NeuralTS, the approximation error for $h(x) - f_1(x)$ includes three parts. The first part is caused by ridge regression. The second part of the approximation error is caused by the distance between ridge regression and Neural Tangent Kernel (NTK). The third part of the approximation error is caused by the distance between NTK and the network function. Because they use the upper bound to make selections, the errors inherently exist in their algorithms.



Figure 6: Two types of exploration: Upward exploration and Downward exploration. f_1 is the exploitation network (estimated reward) and h is the expected reward.

The two types of exploration are described by Figure 6. When the estimated reward is larger than the expected reward, i.e., $h(x) - f_1(x) < 0$, we need to do the ‘downward

exploration’, i.e., lowering the exploration score of x to reduce its chance of being explored; when $h(x) - f_1(x) > 0$, we should do the ‘upward exploration’, i.e., raising the exploration score of x to increase its chance of being explored. For EE-Net, f_2 is to learn $h(x) - f_1(x)$. When $h(x) - f_1(x) > 0$, $f_2(x)$ will also be positive to make the upward exploration. When $h(x) - f_1(x) < 0$, $f_2(x)$ will be negative to make the downward exploration.

A.5 Discussion

As f_2 is randomly initialized, it initially facilitates random exploration by estimating the potential gain of each unseen arm arbitrarily. When an arm is selected and its corresponding reward is observed, f_2 refines its potential gain estimation using the internal state (gradient) of f_1 . In subsequent rounds, when f_1 encounters similar internal states (e.g., facing uncertainty in selecting arms), f_2 can provide more informed potential gain estimates to guide exploration. Over time, as the learner accumulates more internal state and potential gain pairs, f_2 gains a better understanding of various states of f_1 , enabling more accurate potential gain estimation. Consequently, in each round, the combination of f_1 and f_2 computes an exploitation-exploration score for each arm, selecting the optimal one.

In summary, during the initial phase, due to its random initialization, f_2 may produce inaccurate potential gain estimates, leading to suboptimal actions. However, it progressively improves by learning from these suboptimal decisions. As more observations are collected, f_2 refines its estimations, ultimately enhancing decision-making performance. Nevertheless, optimizing the initialization and training strategy of f_2 across different phases remains an interesting direction for future exploration.

Appendix B. Proof of Theorem 5

B.1 Bounds for Generic Neural Networks

In this section, we provide some base lemmas for the neural networks with respect to gradient or loss. Let $\mathbf{x}_t = \mathbf{x}_{t,\hat{i}}$. Recall that $\mathcal{L}_t(\boldsymbol{\theta}) = (f(\mathbf{x}_t; \boldsymbol{\theta}) - r_t)^2/2$. Following (Allen-Zhu et al., 2019; Cao and Gu, 2019), given an instance \mathbf{x} , $\|\mathbf{x}\| = 1$, we define the outputs of hidden layers of the neural network (Eq. (5.1)):

$$\Gamma_0 = \mathbf{x}, \Gamma_l = \sigma(\mathbf{W}_l \mathbf{h}_{l-1}), l \in [L - 1].$$

Then, we define the binary diagonal matrix functioning as ReLU:

$$\mathbf{D}_l = \text{diag}(\mathbb{1}\{(\mathbf{W}_l \Gamma_{l-1})_1\}, \dots, \mathbb{1}\{(\mathbf{W}_l \Gamma_{l-1})_m\}), l \in [L - 1],$$

where $\mathbb{1}$ is the indicator function : $\mathbb{1}(x) = 1$ if $x > 0$; $\mathbb{1}(x) = 0$, otherwise. Accordingly, the neural network (Eq. (5.1)) is represented by

$$f(\mathbf{x}; \boldsymbol{\theta}^1 \text{ or } \boldsymbol{\theta}^2) = \mathbf{W}_L \left(\prod_{l=1}^{L-1} \mathbf{D}_l \mathbf{W}_l \right) \mathbf{x}, \quad (\text{B.1})$$

and

$$\nabla_{\mathbf{W}_l} f = \begin{cases} [\Gamma_{l-1} \mathbf{W}_L (\prod_{\tau=l+1}^{L-1} \mathbf{D}_\tau \mathbf{W}_\tau)]^\top, l \in [L - 1] \\ \Gamma_{L-1}^\top, l = L. \end{cases} \quad (\text{B.2})$$

Then, we have the following auxiliary lemmas.

Lemma 12 *Suppose m, η_1, η_2 satisfy the conditions in Theorem 5. With probability at least $1 - \mathcal{O}(TnL^2) \cdot \exp(-\Omega(m\omega^{2/3}L))$ over the random initialization, for all $t \in [T], i \in [n]$, $\boldsymbol{\theta}$ satisfying $\|\boldsymbol{\theta} - \boldsymbol{\theta}_0\|_2 \leq \omega$ with $\omega \leq \mathcal{O}(L^{-9/2}[\log m]^{-3})$, it holds uniformly that*

$$\begin{aligned} (1) & |f(\mathbf{x}_{t,i}; \boldsymbol{\theta})| \leq \mathcal{O}(1) \\ (2) & \|\nabla_{\boldsymbol{\theta}} f(\mathbf{x}_{t,i}; \boldsymbol{\theta})\|_2 \leq \mathcal{O}(\sqrt{L}) \\ (3) & \|\nabla_{\boldsymbol{\theta}} \mathcal{L}_t(\boldsymbol{\theta})\|_2 \leq \mathcal{O}(\sqrt{L}) \end{aligned}$$

Proof (1) is a simple application of Cauchy–Schwarz inequality.

$$\begin{aligned} |f(\mathbf{x}_{t,i}; \boldsymbol{\theta})| &= |\mathbf{W}_L(\prod_{l=1}^{L-1} \mathbf{D}_l \mathbf{W}_l) \mathbf{x}_{t,i}| \\ &\leq \underbrace{\|\mathbf{W}_L(\prod_{l=1}^{L-1} \mathbf{D}_l \mathbf{W}_l)\|_2}_{I_1} \|\mathbf{x}_{t,i}\|_2 \\ &\leq \mathcal{O}(1) \end{aligned}$$

where I_1 is based on the Lemma B.2 (Cao and Gu, 2019): $I_1 \leq \mathcal{O}(1)$, and $\|\mathbf{x}_{t,i}\|_2 = 1$.

For (2), it holds uniformly that

$$\|\nabla_{\boldsymbol{\theta}} f(\mathbf{x}_{t,i}; \boldsymbol{\theta})\|_2 = \|\text{vec}(\nabla_{\mathbf{W}_1} f)^\top, \dots, \text{vec}(\nabla_{\mathbf{W}_L} f)^\top\|_2 \leq \mathcal{O}(\sqrt{L})$$

where $\|\nabla_{\mathbf{W}_l} f\|_F \leq \mathcal{O}(1), l \in [L]$ is an application of Lemma B.3 (Cao and Gu, 2019) by removing \sqrt{m} .

For (3), we have $\|\nabla_{\boldsymbol{\theta}} \mathcal{L}_t(\boldsymbol{\theta})\|_2 \leq |\mathcal{L}'_t| \cdot \|\nabla_{\boldsymbol{\theta}} f(\mathbf{x}_{t,i}; \boldsymbol{\theta})\|_2 \leq \mathcal{O}(\sqrt{L})$. ■

Lemma 13 *Suppose m, η_1, η_2 satisfy the conditions in Theorem 5. With probability at least $1 - \mathcal{O}(TnL^2) \cdot \exp(-\Omega(m\omega^{2/3}L))$ over the random initialization, for all $\|\mathbf{x}\|_2 = 1$, $\boldsymbol{\theta}$ satisfying $\|\boldsymbol{\theta} - \boldsymbol{\theta}_0\|_2, \|\boldsymbol{\theta}' - \boldsymbol{\theta}_0\|_2 \leq \omega$ with $\omega \leq \mathcal{O}(L^{-9/2}[\log m]^{-3})$, it holds uniformly that*

$$|f(\mathbf{x}; \boldsymbol{\theta}) - f(\mathbf{x}; \boldsymbol{\theta}') - \langle \nabla_{\boldsymbol{\theta}'} f(\mathbf{x}; \boldsymbol{\theta}'), \boldsymbol{\theta} - \boldsymbol{\theta}' \rangle| \leq \mathcal{O}(w^{1/3}L^2\sqrt{\log m})\|\boldsymbol{\theta} - \boldsymbol{\theta}'\|_2.$$

Proof Based on Lemma 4.1 (Cao and Gu, 2019), it holds uniformly that

$$|\sqrt{m}f(\mathbf{x}; \boldsymbol{\theta}) - \sqrt{m}f(\mathbf{x}; \boldsymbol{\theta}') - \langle \sqrt{m}\nabla_{\boldsymbol{\theta}'} f(\mathbf{x}; \boldsymbol{\theta}'), \boldsymbol{\theta} - \boldsymbol{\theta}' \rangle| \leq \mathcal{O}(w^{1/3}L^2\sqrt{m\log(m)})\|\boldsymbol{\theta} - \boldsymbol{\theta}'\|_2,$$

where \sqrt{m} comes from the different scaling of neural network structure. Removing \sqrt{m} completes the proof. ■

Lemma 14 *Suppose m, η_1, η_2 satisfy the conditions in Theorem 5. With probability at least $1 - \mathcal{O}(TnL^2) \cdot \exp(-\Omega(m\omega^{2/3}L))$ over the random initialization, for all $t \in [T], i \in [n]$, $\boldsymbol{\theta}, \boldsymbol{\theta}' \in B(\boldsymbol{\theta}_0, \omega)$ with $\omega \leq \mathcal{O}(L^{-9/2}[\log m]^{-3})$, it holds uniformly that*

$$|f(\mathbf{x}_{t,i}; \boldsymbol{\theta}) - f(\mathbf{x}_{t,i}; \boldsymbol{\theta}')| \leq \mathcal{O}(\omega\sqrt{L}) + \mathcal{O}(\omega^{4/3}L^2\sqrt{\log m}) \quad (\text{B.3})$$

Proof Based on Lemma 13, we have

$$\begin{aligned}
 & |f(\mathbf{x}_{t,i}; \boldsymbol{\theta}) - f(\mathbf{x}_{t,i}; \boldsymbol{\theta}')| \\
 & \stackrel{(a)}{\leq} |\langle \nabla_{\boldsymbol{\theta}'} f(\mathbf{x}_{t,i}; \boldsymbol{\theta}'), \boldsymbol{\theta} - \boldsymbol{\theta}' \rangle| + \mathcal{O}(\omega^{1/3} L^2 \sqrt{\log m}) \|\boldsymbol{\theta} - \boldsymbol{\theta}'\|_2 \\
 & \stackrel{(b)}{\leq} \|\nabla_{\boldsymbol{\theta}'} f(\mathbf{x}_{t,i}; \boldsymbol{\theta}')\|_2 \cdot \|\boldsymbol{\theta} - \boldsymbol{\theta}'\|_2 + \mathcal{O}(\omega^{1/3} L^2 \sqrt{\log m}) \|\boldsymbol{\theta} - \boldsymbol{\theta}'\|_2 \\
 & \stackrel{(c)}{\leq} \mathcal{O}(\sqrt{L}) \|\boldsymbol{\theta} - \boldsymbol{\theta}'\|_2 + \mathcal{O}(\omega^{1/3} L^2 \sqrt{\log m}) \|\boldsymbol{\theta} - \boldsymbol{\theta}'\|_2 \\
 & \leq \mathcal{O}(\omega \sqrt{L}) + \mathcal{O}(\omega^{4/3} L^2 \sqrt{\log m}),
 \end{aligned}$$

where (a) is an application of Lemma 13, (b) is based on the Cauchy–Schwarz inequality, and (c) is due to Lemma 12. The proof is completed. \blacksquare

Lemma 15 (Almost Convexity of Loss) *Let $\mathcal{L}_t(\boldsymbol{\theta}) = (\sqrt{m}f(\mathbf{x}_t; \boldsymbol{\theta}) - r_t)^2/2$. Suppose m, η_1, η_2 satisfy the conditions in Theorem 5. With probability at least $1 - \mathcal{O}(TnL^2) \exp[-\Omega(m\omega^{2/3}L)]$ over randomness, for all $t \in [T], i \in [n]$, and $\boldsymbol{\theta}, \boldsymbol{\theta}'$ satisfying $\|\boldsymbol{\theta} - \boldsymbol{\theta}_0\|_2 \leq \omega$ and $\|\boldsymbol{\theta}' - \boldsymbol{\theta}_0\|_2 \leq \omega$ with $\omega \leq \mathcal{O}(L^{-6}[\log m]^{-3/2})$, it holds uniformly that*

$$\mathcal{L}_t(\boldsymbol{\theta}') \geq \mathcal{L}_t(\boldsymbol{\theta}) + \langle \nabla_{\boldsymbol{\theta}} \mathcal{L}_t(\boldsymbol{\theta}), \boldsymbol{\theta}' - \boldsymbol{\theta} \rangle - \epsilon.$$

where $\epsilon = \mathcal{O}(\omega^{4/3} L^3 \sqrt{\log m})$

Proof Let \mathcal{L}'_t be the derivative of \mathcal{L}_t with respect to $f(\mathbf{x}_{t,i}; \boldsymbol{\theta})$. Then, it holds that $|\mathcal{L}'_t| \leq \mathcal{O}(1)$ based on Lemma 12. Then, by convexity of \mathcal{L}_t , we have

$$\begin{aligned}
 & \mathcal{L}_t(\boldsymbol{\theta}') - \mathcal{L}_t(\boldsymbol{\theta}) \\
 & \stackrel{(a)}{\geq} \mathcal{L}'_t[f(\mathbf{x}_{t,i}; \boldsymbol{\theta}') - f(\mathbf{x}_{t,i}; \boldsymbol{\theta})] \\
 & \stackrel{(b)}{\geq} \mathcal{L}'_t \langle \nabla f(\mathbf{x}_{t,i}; \boldsymbol{\theta}), \boldsymbol{\theta}' - \boldsymbol{\theta} \rangle \\
 & \quad - |\mathcal{L}'_t| \cdot |f(\mathbf{x}_{t,i}; \boldsymbol{\theta}') - f(\mathbf{x}_{t,i}; \boldsymbol{\theta}) - \langle \nabla f(\mathbf{x}_{t,i}; \boldsymbol{\theta}), \boldsymbol{\theta}' - \boldsymbol{\theta} \rangle| \\
 & \geq \langle \nabla_{\boldsymbol{\theta}} \mathcal{L}_t(\boldsymbol{\theta}), \boldsymbol{\theta}' - \boldsymbol{\theta} \rangle - |\mathcal{L}'_t| \cdot |f(\mathbf{x}_{t,i}; \boldsymbol{\theta}') - f(\mathbf{x}_{t,i}; \boldsymbol{\theta}) - \langle \nabla f(\mathbf{x}_{t,i}; \boldsymbol{\theta}), \boldsymbol{\theta}' - \boldsymbol{\theta} \rangle| \\
 & \stackrel{(c)}{\geq} \langle \nabla_{\boldsymbol{\theta}'} \mathcal{L}_t, \boldsymbol{\theta}' - \boldsymbol{\theta} \rangle - \mathcal{O}(\omega^{4/3} L^3 \sqrt{\log m}) \\
 & \geq \langle \nabla_{\boldsymbol{\theta}'} \mathcal{L}_t, \boldsymbol{\theta}' - \boldsymbol{\theta} \rangle - \epsilon
 \end{aligned}$$

where (a) is due to the convexity of \mathcal{L}_t , (b) is an application of triangle inequality, and (c) is the application of Lemma 13. The proof is completed. \blacksquare

Lemma 16 (Trajectory Ball) *Suppose m, η_1, η_2 satisfy the conditions in Theorem 5. With probability at least $1 - \mathcal{O}(TnL^2) \exp[-\Omega(m\omega^{2/3}L)]$ over randomness of $\boldsymbol{\theta}_0$, for any $R > 0$, it holds uniformly that*

$$\|\boldsymbol{\theta}_t^1 - \boldsymbol{\theta}_0\|_2 \leq \mathcal{O}(R) \ \& \ \|\boldsymbol{\theta}_t^2 - \boldsymbol{\theta}_0\|_2 \leq \mathcal{O}(R), t \in [T].$$

Proof Let $\omega = \Omega(R)$. The proof follows a simple induction. Obviously, $\boldsymbol{\theta}_0$ is in $B(\boldsymbol{\theta}_0, \omega)$. Suppose that $\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \dots, \boldsymbol{\theta}_T \in \mathcal{B}(\boldsymbol{\theta}_0^2, \omega)$. We have, for any $t \in [T]$,

$$\begin{aligned} \|\boldsymbol{\theta}_T - \boldsymbol{\theta}_0\|_2 &\leq \sum_{t=1}^T \|\boldsymbol{\theta}_t - \boldsymbol{\theta}_{t-1}\|_2 \leq \sum_{t=1}^T \eta \|\nabla \mathcal{L}_t(\boldsymbol{\theta}_{t-1})\| \leq \sum_{t=1}^T \eta \mathcal{O}(\sqrt{L}) \\ &= \mathcal{O}(TR^2\sqrt{L}/\sqrt{m}) \leq \mathcal{O}(R) \end{aligned}$$

The proof is completed. ■

Theorem 17 (Instance-dependent Loss Bound) *Let $\mathcal{L}_t(\boldsymbol{\theta}) = (f(\mathbf{x}_{t,\hat{i}}; \boldsymbol{\theta}) - r_{t,\hat{i}})^2/2$. Suppose m, η_1, η_2 satisfy the conditions in Theorem 5. With probability at least $1 - \mathcal{O}(TnL^2) \exp[-\Omega(m\omega^{2/3}L)]$ over randomness of $\boldsymbol{\theta}_0^2$, given any $R > 0$ it holds that*

$$\sum_{t=1}^T \mathcal{L}_t(\boldsymbol{\theta}_{t-1}^2) \leq \sum_{t=1}^T \mathcal{L}_t(\boldsymbol{\theta}^*) + \mathcal{O}(1) + \frac{TLR^2}{\sqrt{m}} + \mathcal{O}\left(\frac{TR^{4/3}L^2\sqrt{\log m}}{m^{1/3}}\right). \quad (\text{B.4})$$

where $\boldsymbol{\theta}^* = \arg \inf_{\boldsymbol{\theta} \in B(\boldsymbol{\theta}_0^2, R)} \sum_{t=1}^T \mathcal{L}_t(\boldsymbol{\theta})$.

Proof Let $\boldsymbol{\theta}' \in B(\boldsymbol{\theta}_0^2, R)$. In round t , based on Lemma 16, for all $t \in [T]$, $\|\boldsymbol{\theta}_t - \boldsymbol{\theta}'\|_2 \leq \omega = \Omega(R)$, it holds uniformly,

$$\mathcal{L}_t(\boldsymbol{\theta}_{t-1}^2) - \mathcal{L}_t(\boldsymbol{\theta}') \leq \langle \nabla \mathcal{L}_t(\boldsymbol{\theta}_{t-1}^2), \boldsymbol{\theta}_{t-1}^2 - \boldsymbol{\theta}' \rangle + \epsilon,$$

where $\epsilon = \mathcal{O}(\omega^{4/3}L^2\sqrt{\log m})$.

Therefore, for all $t \in [T]$, $\boldsymbol{\theta}' \in B(\boldsymbol{\theta}_0^2, R)$, it holds uniformly

$$\begin{aligned} \mathcal{L}_t(\boldsymbol{\theta}_{t-1}^2) - \mathcal{L}_t(\boldsymbol{\theta}') &\stackrel{(a)}{\leq} \frac{\langle \boldsymbol{\theta}_{t-1}^2 - \boldsymbol{\theta}_t^2, \boldsymbol{\theta}_{t-1}^2 - \boldsymbol{\theta}' \rangle}{\eta_2} + \epsilon \\ &\stackrel{(b)}{=} \frac{\|\boldsymbol{\theta}_{t-1}^2 - \boldsymbol{\theta}'\|_2^2 + \|\boldsymbol{\theta}_{t-1}^2 - \boldsymbol{\theta}_t^2\|_2^2 - \|\boldsymbol{\theta}_t^2 - \boldsymbol{\theta}'\|_2^2}{2\eta_2} + \epsilon \\ &\stackrel{(c)}{\leq} \frac{\|\boldsymbol{\theta}_{t-1}^2 - \boldsymbol{\theta}'\|_2^2 - \|\boldsymbol{\theta}_t^2 - \boldsymbol{\theta}'\|_2^2}{2\eta_2} + \mathcal{O}(L\eta_2) + \epsilon \end{aligned}$$

where (a) is because of the definition of gradient descent, (b) is due to the fact $2\langle A, B \rangle = \|A\|_F^2 + \|B\|_F^2 - \|A - B\|_F^2$, (c) is by $\|\boldsymbol{\theta}_{t-1}^2 - \boldsymbol{\theta}_t^2\|_2^2 = \|\eta_2 \nabla_{\boldsymbol{\theta}_{t-1}^2} \mathcal{L}_t(\boldsymbol{\theta}_{t-1}^2)\|_2^2 \leq \mathcal{O}(\eta_2^2 L)$.

Then, for T rounds, we have

$$\begin{aligned}
 & \sum_{t=1}^T \mathcal{L}_t(\boldsymbol{\theta}_{t-1}) - \sum_{t=1}^T \mathcal{L}_t(\boldsymbol{\theta}') \\
 \stackrel{(a)}{\leq} & \frac{\|\boldsymbol{\theta}_0^2 - \boldsymbol{\theta}'\|_2^2}{2\eta_2} + \sum_{t=2}^T \|\boldsymbol{\theta}_{t-1}^2 - \boldsymbol{\theta}'\|_2^2 \left(\frac{1}{2\eta_2} - \frac{1}{2\eta_2} \right) + \sum_{t=1}^T L\eta_2 + T\epsilon \\
 \leq & \frac{\|\boldsymbol{\theta}_0^2 - \boldsymbol{\theta}'\|_2^2}{2\eta_2} + \sum_{t=1}^T L\eta_2 + T\epsilon \\
 \stackrel{(b)}{\leq} & \mathcal{O}\left(\frac{R^2}{\sqrt{m}\eta_2}\right) + \sum_{t=1}^T L\eta_2 + T\epsilon \\
 \stackrel{(c)}{\leq} & \mathcal{O}(1) + \frac{TLR^2}{\sqrt{m}} + \mathcal{O}\left(\frac{TR^{4/3}L^2\sqrt{\log m}}{m^{1/3}}\right)
 \end{aligned}$$

where (a) is by simply discarding the last term and (b) is because both $\boldsymbol{\theta}_0^2$ and $\boldsymbol{\theta}'$ are in the ball $B(\boldsymbol{\theta}_0^2, R)$, and (c) is by $\eta_2 = \frac{R^2}{\sqrt{m}}$ and replacing ϵ with $\omega = \mathcal{O}(R)$. The proof is completed. \blacksquare

B.2 Exploration Error Bound

Lemma 18 *Suppose m, η_1, η_2 satisfies the conditions in Theorem 5. In round $t \in [T]$, let*

$$\hat{i} = \arg \max_{i \in [k]} \left(f_1(\mathbf{x}_{t,i}; \boldsymbol{\theta}_{t-1}^1) + f_2(\phi(\mathbf{x}_{t,i}); \boldsymbol{\theta}_{t-1}^2) \right),$$

and denote the policy by π_t . Then, for any $\delta \in (0, 1)$, $R > 0$, with probability at least $1 - \delta$, for $t \in [T]$, it holds uniformly

$$\begin{aligned}
 & \frac{1}{t} \sum_{\tau=1}^t \mathbb{E}_{r_{\tau,\hat{i}}} \left[\left| f_1(\mathbf{x}_{\tau,\hat{i}}; \boldsymbol{\theta}_{\tau-1}^1) + f_2(\phi(\mathbf{x}_{\tau,\hat{i}}); \boldsymbol{\theta}_{\tau-1}^2) - r_{\tau,\hat{i}} \right| \mid \pi_t, \mathcal{H}_{\tau-1} \right] \\
 & \leq \frac{\sqrt{\Psi(\boldsymbol{\theta}_0^2, R) + \mathcal{O}(1)}}{\sqrt{t}} + \sqrt{\frac{2 \log(\mathcal{O}(1)/\delta)}{t}}.
 \end{aligned} \tag{5.5}$$

where $\mathcal{H}_t = \{\mathbf{x}_{\tau,\hat{i}}, r_{\tau,\hat{i}}\}_{\tau=1}^t$ represents of historical data selected by π_τ and expectation is taken over the reward.

Proof First, according to Lemma 16, $\boldsymbol{\theta}_0^2, \dots, \boldsymbol{\theta}_{T-1}^2$ all are in $\mathcal{B}(\boldsymbol{\theta}_0, R)$. Then, according to Lemma 12, for any $\mathbf{x} \in \mathbb{R}^d$, $\|\mathbf{x}\|_2 = 1$, it holds uniformly $|f_1(\mathbf{x}_{t,\hat{i}}; \boldsymbol{\theta}_t^1) + f_2(\phi(\mathbf{x}_{t,\hat{i}}); \boldsymbol{\theta}_t^2) - r_{t,\hat{i}}| \leq \mathcal{O}(1)$.

Then, for any $\tau \in [t]$, define

$$\begin{aligned}
 V_\tau := & \mathbb{E}_{r_{\tau,\hat{i}}} \left[\left| f_2(\phi(\mathbf{x}_{\tau,\hat{i}}); \boldsymbol{\theta}_{\tau-1}^2) - (r_{\tau,\hat{i}} - f_1(\mathbf{x}_{\tau,\hat{i}}; \boldsymbol{\theta}_{\tau-1}^1)) \right| \right] \\
 & - \left| f_2(\phi(\mathbf{x}_{\tau,\hat{i}}); \boldsymbol{\theta}_{\tau-1}^2) - (r_{\tau,\hat{i}} - f_1(\mathbf{x}_{\tau,\hat{i}}; \boldsymbol{\theta}_{\tau-1}^1)) \right|
 \end{aligned} \tag{B.5}$$

Then, we have

$$\begin{aligned} \mathbb{E}[V_\tau | F_{\tau-1}] &= \mathbb{E}_{r_{\tau,\hat{i}}} \left[|f_2(\phi(\mathbf{x}_{\tau,\hat{i}}); \boldsymbol{\theta}_{\tau-1}^2) - (r_{\tau,\hat{i}} - f_1(\mathbf{x}_{\tau,\hat{i}}; \boldsymbol{\theta}_{\tau-1}^1))| \right] \\ &\quad - \mathbb{E}_{r_{\tau,\hat{i}}} \left[|f_2(\phi(\mathbf{x}_{\tau,\hat{i}}); \boldsymbol{\theta}_{\tau-1}^2) - (r_{\tau,\hat{i}} - f_1(\mathbf{x}_{\tau,\hat{i}}; \boldsymbol{\theta}_{\tau-1}^1))| \right] \\ &= 0 \end{aligned} \quad (\text{B.6})$$

where $F_{\tau-1}$ denotes the σ -algebra generated by the history $\mathcal{H}_{\tau-1}$.

Therefore, the sequence $\{V_\tau\}_{\tau=1}^t$ is the martingale difference sequence. Applying the Hoeffding-Azuma inequality, with probability at least $1 - \delta$, we have

$$\mathbb{P} \left[\frac{1}{t} \sum_{\tau=1}^t V_\tau - \underbrace{\frac{1}{t} \sum_{\tau=1}^t \mathbb{E}[V_\tau | \mathbf{F}_{\tau-1}]}_{I_1} > \sqrt{\frac{2 \log(1/\delta)}{t}} \right] \leq \delta \quad (\text{B.7})$$

As I_1 is equal to 0, we have

$$\begin{aligned} &\frac{1}{t} \sum_{\tau=1}^t \mathbb{E}_{r_{\tau,\hat{i}}} \left[|f_2(\phi(\mathbf{x}_{\tau,\hat{i}}); \boldsymbol{\theta}_{\tau-1}^2) - (r_{\tau,\hat{i}} - f_1(\mathbf{x}_{\tau,\hat{i}}; \boldsymbol{\theta}_{\tau-1}^1))| \right] \\ &\leq \underbrace{\frac{1}{t} \sum_{\tau=1}^t |f_2(\phi(\mathbf{x}_{\tau,\hat{i}}); \boldsymbol{\theta}_{\tau-1}^2) - (r_{\tau,\hat{i}} - f_1(\mathbf{x}_{\tau,\hat{i}}; \boldsymbol{\theta}_{\tau-1}^1))|}_{I_3} + \sqrt{\frac{2 \log(1/\delta)}{t}}. \end{aligned} \quad (\text{B.8})$$

For I_3 , based on Theorem 17, for any $\boldsymbol{\theta}'$ satisfying $\|\boldsymbol{\theta}' - \boldsymbol{\theta}_0^2\|_2 \leq R/m^{1/4}$, with probability at least $1 - \delta$, we have

$$\begin{aligned} I_3 &\leq \frac{1}{t} \sqrt{t} \sqrt{\sum_{\tau=1}^t \left(f_2(\phi(\mathbf{x}_{\tau,\hat{i}}); \boldsymbol{\theta}_{\tau-1}^2) - (r_{\tau,\hat{i}} - f_1(\mathbf{x}_{\tau,\hat{i}}; \boldsymbol{\theta}_{\tau-1}^1)) \right)^2} \\ &\leq \frac{1}{t} \sqrt{t} \sqrt{\sum_{\tau=1}^t \left(f_2(\phi(\mathbf{x}_{\tau,\hat{i}}); \boldsymbol{\theta}') - (r_{\tau,\hat{i}} - f_1(\mathbf{x}_{\tau,\hat{i}}; \boldsymbol{\theta}_{\tau-1}^1)) \right)^2} + \frac{\mathcal{O}(1)}{\sqrt{t}} \\ &\stackrel{(a)}{\leq} \frac{\sqrt{\Psi(\boldsymbol{\theta}_0^2, R)} + \mathcal{O}(1)}{\sqrt{t}}. \end{aligned} \quad (\text{B.9})$$

where (a) is based on the definition of instance-dependent complexity term. Combining the above inequalities together, with probability at least $1 - \delta$, we have

$$\begin{aligned} &\frac{1}{t} \sum_{\tau=1}^t \mathbb{E}_{r_{\tau,\hat{i}}} \left[|f_2(\phi(\mathbf{x}_{\tau,\hat{i}}); \boldsymbol{\theta}_{\tau-1}^2) - (r_{\tau,\hat{i}} - f_1(\mathbf{x}_{\tau,\hat{i}}; \boldsymbol{\theta}_{\tau-1}^1))| \right] \\ &\leq \frac{\sqrt{\Psi(\boldsymbol{\theta}_0^2, R)} + \mathcal{O}(1)}{\sqrt{t}} + \sqrt{\frac{2 \log(\mathcal{O}(1)/\delta)}{t}}. \end{aligned} \quad (\text{B.10})$$

The proof is completed. ■

Corollary 19 *Suppose m, η_1, η_2 satisfy the conditions in Theorem 5. For any $t \in [T]$, let*

$$i^* = \arg \max_{i \in [n]} [h(\mathbf{x}_{t,i})],$$

and r_{t,i^*} is the corresponding reward, and denote the policy by π^* . Let $\boldsymbol{\theta}_{t-1}^{1,*}, \boldsymbol{\theta}_{t-1}^{2,*}$ be the intermediate parameters trained by Algorithm 1 using the data select by π^* . Then, with probability at least $(1 - \delta)$ over the random of the initialization, for any $\delta \in (0, 1), R > 0$, it holds that

$$\begin{aligned} \frac{1}{t} \sum_{\tau=1}^t \mathbb{E}_{r_{\tau,i^*}} \left[\left| f_2(\phi(\mathbf{x}_{\tau,i^*}); \boldsymbol{\theta}_{\tau-1}^{2,*}) - \left(r_{\tau,i^*} - f_1(\mathbf{x}_{\tau,i^*}; \boldsymbol{\theta}_{\tau-1}^{1,*}) \right) \right| \mid \pi^*, \mathcal{H}_{\tau-1}^* \right] \\ \leq \frac{\sqrt{\Psi(\boldsymbol{\theta}_0^2, R) + \mathcal{O}(1)}}{\sqrt{t}} + \sqrt{\frac{2 \log(\mathcal{O}(1)/\delta)}{t}}, \end{aligned} \quad (\text{B.11})$$

where $\mathcal{H}_{\tau-1}^* = \{\mathbf{x}_{\tau',i^*}, r_{\tau',i^*}\}_{\tau'=1}^{\tau-1}$ represents the historical data produced by π^* and the expectation is taken over the reward.

Proof This a direct corollary of Lemma 8, given the optimal historical pairs $\{\mathbf{x}_{\tau',i^*}, r_{\tau',i^*}\}_{\tau'=1}^{t-1}$ according to π^* . For brevity, let $f_2(\phi(\mathbf{x}); \boldsymbol{\theta}_\tau^{2,*})$ represent $f_2(\nabla_{\boldsymbol{\theta}_\tau^{1,*}} f_1(\mathbf{x}; \boldsymbol{\theta}_\tau^{1,*}); \boldsymbol{\theta}_\tau^{2,*})$.

Suppose that, for each $\tau \in [t-1]$, $\boldsymbol{\theta}_\tau^{1,*}$ and $\boldsymbol{\theta}_\tau^{2,*}$ are the parameters training on $\{\mathbf{x}_{\tau'}, r_{\tau'}^*\}_{\tau'=1}^\tau$ according to Algorithm 1 according to π^* . Note that these pairs $\{\mathbf{x}_{\tau'}, r_{\tau'}^*\}_{\tau'=1}^\tau$ are unknown to the algorithm we run, and the parameters $(\boldsymbol{\theta}_\tau^{1,*}, \boldsymbol{\theta}_\tau^{2,*})$ are not estimated. However, for the analysis, it is sufficient to show that there exist such parameters so that the conditional expectation of the error can be bounded.

Then, we define

$$\begin{aligned} V_\tau := \mathbb{E}_{r_{\tau,i^*}} \left[\left| f_2(\phi(\mathbf{x}_{\tau,i^*}); \boldsymbol{\theta}_{\tau-1}^{2,*}) - \left(r_{\tau,i^*} - f_1(\mathbf{x}_{\tau,i^*}; \boldsymbol{\theta}_{\tau-1}^{1,*}) \right) \right| \right] \\ - \left| f_2(\phi(\mathbf{x}_{\tau,i^*}); \boldsymbol{\theta}_{\tau-1}^{2,*}) - \left(r_{\tau,i^*} - f_1(\mathbf{x}_{\tau,i^*}; \boldsymbol{\theta}_{\tau-1}^{1,*}) \right) \right|. \end{aligned} \quad (\text{B.12})$$

Then, taking the expectation over reward, we have

$$\begin{aligned} \mathbb{E}[V_\tau | \mathbf{F}_{\tau-1}] &= \mathbb{E}_{r_{\tau,i^*}} \left[\left| f_2(\phi(\mathbf{x}_{\tau,i^*}); \boldsymbol{\theta}_{\tau-1}^{2,*}) - \left(r_{\tau,i^*} - f_1(\mathbf{x}_{\tau,i^*}; \boldsymbol{\theta}_{\tau-1}^{1,*}) \right) \right| \right] \\ &\quad - \mathbb{E}_{r_{\tau,i^*}} \left[\left| f_2(\phi(\mathbf{x}_{\tau,i^*}); \boldsymbol{\theta}_{\tau-1}^{2,*}) - \left(r_{\tau,i^*} - f_1(\mathbf{x}_{\tau,i^*}; \boldsymbol{\theta}_{\tau-1}^{1,*}) \right) \right| \mid \mathbf{F}_{\tau-1} \right] \\ &= 0, \end{aligned} \quad (\text{B.13})$$

where $\mathbf{F}_{\tau-1}$ denotes the σ -algebra generated by the history $\mathcal{H}_{\tau-1}^* = \{\mathbf{x}_{\tau',i^*}, r_{\tau',i^*}\}_{\tau'=1}^{\tau-1}$.

Therefore, $\{V_\tau\}_{\tau=1}^t$ is a martingale difference sequence. Similarly to Lemma 8, applying the Hoeffding-Azuma inequality to V_τ , with probability $1 - \delta$, we have

$$\begin{aligned}
 & \frac{1}{t} \sum_{\tau=1}^t \mathbb{E} \left[\left| f_2(\phi(\mathbf{x}_{\tau, i^*}); \boldsymbol{\theta}_{\tau-1}^{2,*}) - \left(r_{\tau, i^*} - f_1(\mathbf{x}_{\tau, i^*}; \boldsymbol{\theta}_{\tau-1}^{1,*}) \right) \right| \right] \\
 & \leq \frac{1}{t} \sum_{\tau=1}^t \left| f_2(\phi(\mathbf{x}_{\tau, i^*}); \boldsymbol{\theta}_{\tau-1}^{2,*}) - \left(r_{\tau, i^*} - f_1(\mathbf{x}_{\tau, i^*}; \boldsymbol{\theta}_{\tau-1}^{1,*}) \right) \right| + \sqrt{\frac{2 \log(1/\delta)}{t}} \\
 & \leq \frac{1}{t} \sqrt{t} \sqrt{\sum_{\tau=1}^t \left(f_2(\phi(\mathbf{x}_{\tau, i^*}); \boldsymbol{\theta}^{2,*}) - \left(r_{\tau, i^*} - f_1(\mathbf{x}_{\tau, i^*}; \boldsymbol{\theta}_{\tau-1}^{1,*}) \right) \right)^2} + \sqrt{\frac{2 \log(1/\delta)}{t}} \quad (\text{B.14}) \\
 & \stackrel{(a)}{\leq} \frac{1}{t} \sqrt{t} \sqrt{\sum_{\tau=1}^t \left(f_2(\phi(\mathbf{x}_{\tau, i^*}); \boldsymbol{\theta}') - \left(r_{\tau, i^*} - f_1(\mathbf{x}_{\tau, i^*}; \boldsymbol{\theta}_{\tau-1}^{1,*}) \right) \right)^2} + \frac{\mathcal{O}(1)}{\sqrt{t}} + \sqrt{\frac{2 \log(1/\delta)}{t}} \\
 & \stackrel{(b)}{\leq} \frac{\sqrt{\Psi(\boldsymbol{\theta}_0^2, R)} + \mathcal{O}(1)}{\sqrt{t}} + \sqrt{\frac{2 \log(1/\delta)}{t}},
 \end{aligned}$$

where (a) is an application of Lemma 17 for all $\boldsymbol{\theta}' \in B(\boldsymbol{\theta}_0^2, R)$ and (b) is based on the definition of instance-dependent complexity term. Combining the above inequalities, the proof is complete. \blacksquare

B.3 Main Proof

In this section, we provide the proof of Theorem 5.

Theorem 5 *For any $\delta \in (0, 1)$, $R > 0$, suppose $m \geq \Omega(\text{poly}(T, L, R, n, \log(1/\delta)))$, $\eta_1 = \eta_2 = \frac{R^2}{\sqrt{m}}$. Then, with probability at least $1 - \delta$ over the initialization, the pseudo regret of Algorithm 1 in T rounds satisfies*

$$\mathbf{R}_T \leq \sqrt{T} \cdot \mathcal{O} \left(\sqrt{\Psi(\boldsymbol{\theta}_0^2, R)} + \sqrt{2 \log(1/\delta)} \right) + \mathcal{O}(1). \quad (5.3)$$

Proof For brevity, let $f(\mathbf{x}; \boldsymbol{\theta}_{t-1}) = f_2(\phi(\mathbf{x}); \boldsymbol{\theta}_{t-1}^2) + f_1(\mathbf{x}; \boldsymbol{\theta}_{t-1}^1)$. Then, the pseudo regret of round t is given by

$$\begin{aligned}
 R_t &= h(\mathbf{x}_{t,i^*}) - h(\mathbf{x}_{t,\hat{i}}) \\
 &= \mathbb{E}_{r_{t,i}, i \in [n]} [r_{t,i^*} - r_{t,\hat{i}}] \\
 &= \mathbb{E}_{r_{t,i}, i \in [n]} [r_{t,i^*} - r_{t,\hat{i}}] \\
 &= \mathbb{E}_{r_{t,i}, i \in [n]} [r_{t,i^*} - f(\mathbf{x}_{t,\hat{i}}; \boldsymbol{\theta}_{t-1}) + f(\mathbf{x}_{t,\hat{i}}; \boldsymbol{\theta}_{t-1}) - r_{t,\hat{i}}] \\
 &\leq \mathbb{E}_{r_{t,i}, i \in [n]} \underbrace{[r_{t,i^*} - f(\mathbf{x}_{t,i^*}; \boldsymbol{\theta}_{t-1}) + f(\mathbf{x}_{t,\hat{i}}; \boldsymbol{\theta}_{t-1}) - f(\mathbf{x}_{t,\hat{i}}; \boldsymbol{\theta}_{t-1})]}_{I_1} + f(\mathbf{x}_{t,\hat{i}}; \boldsymbol{\theta}_{t-1}) - r_{t,\hat{i}} \\
 &= \mathbb{E}_{r_{t,i}, i \in [n]} [r_{t,i^*} - f(\mathbf{x}_{t,i^*}; \boldsymbol{\theta}_{t-1}) + f(\mathbf{x}_{t,\hat{i}}; \boldsymbol{\theta}_{t-1}) - r_{t,\hat{i}}] \\
 &\stackrel{(a)}{=} \mathbb{E}_{r_{t,i}, i \in [n]} [r_{t,i^*} - f(\mathbf{x}_{t,i^*}; \boldsymbol{\theta}_{t-1}^*) + f(\mathbf{x}_{t,i^*}; \boldsymbol{\theta}_{t-1}^*) - f(\mathbf{x}_{t,i^*}; \boldsymbol{\theta}_{t-1}) + f(\mathbf{x}_{t,\hat{i}}; \boldsymbol{\theta}_{t-1}) - r_{t,\hat{i}}] \\
 &\leq \mathbb{E}_{r_{t,i}, i \in [n]} \left[\left| f_2(\phi(\mathbf{x}_{t,i^*}); \boldsymbol{\theta}_{t-1}^{2,*}) - (r_{t,i^*} - f_1(\mathbf{x}_{t,i^*}; \boldsymbol{\theta}_{t-1}^{1,*})) \right| \right] \\
 &\quad + \left| f_2(\phi(\mathbf{x}_{t,i^*}); \boldsymbol{\theta}_{t-1}^{2,*}) - f_2(\phi(\mathbf{x}_{t,i^*}); \boldsymbol{\theta}_{t-1}^2) \right| \\
 &\quad + \left| f_1(\mathbf{x}_{t,i^*}; \boldsymbol{\theta}_{t-1}^{1,*}) - f_1(\mathbf{x}_{t,i^*}; \boldsymbol{\theta}_{t-1}^1) \right| \\
 &\quad + \mathbb{E}_{r_{t,i}, i \in [n]} \left[\left| f_2(\phi(\mathbf{x}_{t,\hat{i}}); \boldsymbol{\theta}_{t-1}^2) - (r_{t,\hat{i}} - f_1(\mathbf{x}_{t,\hat{i}}; \boldsymbol{\theta}_{t-1}^1)) \right| \right]
 \end{aligned} \tag{B.15}$$

where I_1 is because $f(\mathbf{x}_{t,\hat{i}}; \boldsymbol{\theta}_{t-1}) = \max_{i \in [n]} f(\mathbf{x}_{t,i}; \boldsymbol{\theta}_{t-1})$ and $f(\mathbf{x}_{t,\hat{i}}; \boldsymbol{\theta}_{t-1}) - f(\mathbf{x}_{t,i^*}; \boldsymbol{\theta}_{t-1}) \geq 0$ and (a) introduces the intermediate parameters $\boldsymbol{\theta}_{t-1}^* = (\boldsymbol{\theta}_{t-1}^{1,*}, \boldsymbol{\theta}_{t-1}^{2,*})$ for analysis, which will be suitably chosen.

Therefore, we have

$$\begin{aligned}
 \mathbf{R}_T &= \sum_{t=1}^T R_t \\
 &\leq \sum_{t=1}^T \mathbb{E}_{r_{t,i}, i \in [n]} \underbrace{\left[\left| f_2(\phi(\mathbf{x}_{t,i^*}); \boldsymbol{\theta}_{t-1}^{2,*}) - (r_{t,i^*} - f_1(\mathbf{x}_{t,i^*}; \boldsymbol{\theta}_{t-1}^{1,*})) \right| \right]}_{I_2} \\
 &\quad + \sum_{t=1}^T \underbrace{\left| f_2(\phi(\mathbf{x}_{t,i^*}); \boldsymbol{\theta}_{t-1}^{2,*}) - f_2(\phi(\mathbf{x}_{t,i^*}); \boldsymbol{\theta}_{t-1}^2) \right|}_{I_3} + \sum_{t=1}^T \underbrace{\left| f_1(\mathbf{x}_{t,i^*}; \boldsymbol{\theta}_{t-1}^{1,*}) - f_1(\mathbf{x}_{t,i^*}; \boldsymbol{\theta}_{t-1}^1) \right|}_{I_4} \\
 &\quad + \sum_{t=1}^T \mathbb{E}_{r_{t,i}, i \in [n]} \underbrace{\left[\left| f_2(\phi(\mathbf{x}_{t,\hat{i}}); \boldsymbol{\theta}_{t-1}^2) - (r_{t,\hat{i}} - f_1(\mathbf{x}_{t,\hat{i}}; \boldsymbol{\theta}_{t-1}^1)) \right| \right]}_{I_5}
 \end{aligned} \tag{B.16}$$

$$\begin{aligned}
 &\leq 2 \sum_{t=1}^T \left(\frac{\sqrt{\Psi(\boldsymbol{\theta}_0^2, R)} + \mathcal{O}(1)}{\sqrt{T}} + \sqrt{\frac{2 \log(\mathcal{O}(1)/\delta)}{T}} \right) \\
 &\quad + 2 \sum_{t=1}^T \left(\mathcal{O}(\sqrt{LR}) + \mathcal{O}(R^{4/3} L^2 \sqrt{\log m/m^{1/3}}) \right) \\
 &\leq \sum_{t=1}^T \left(\frac{2\sqrt{\Psi(\boldsymbol{\theta}_0^2, R)} + \mathcal{O}(1)}{\sqrt{T}} + 2\sqrt{\frac{2 \log(\mathcal{O}(1)/\delta)}{T}} \right) \\
 &\quad + \mathcal{O}(\sqrt{LTR}) + \mathcal{O}(TR^{4/3} L^2 \sqrt{\log m/m^{1/3}}) \\
 &\stackrel{(b)}{\leq} 2\sqrt{\Psi(\boldsymbol{\theta}_0^2, R)T} + 2\sqrt{2 \log(\mathcal{O}(1)/\delta)T} + \mathcal{O}(1)
 \end{aligned} \tag{B.17}$$

I_2 and I_5 is based on Lemma 8 and Corollary 19 , i.e., $I_2, I_5 \leq \frac{\sqrt{\Psi(\boldsymbol{\theta}_0^2, R)} + \mathcal{O}(1)}{\sqrt{T}} + \sqrt{\frac{2 \log(\mathcal{O}(1)/\delta)}{T}}$. $I_3, I_4 \leq \mathcal{O}(1)$ are based on Lemma 14, respectively, because both $\boldsymbol{\theta}_{t-1}^2, \boldsymbol{\theta}_{t-1}^{2,*} \in B(\boldsymbol{\theta}_0^2, R)$ and $\boldsymbol{\theta}_{t-1}^1, \boldsymbol{\theta}_{t-1}^{1,*} \in B(\boldsymbol{\theta}_0^1, R)$. (b) is by the proper choice of m , i.e., when m is large enough, we have $I_3, I_4 \leq \mathcal{O}(1)$.

The proof is completed. \blacksquare

Appendix C. Connections with Neural Tangent Kernel

Lemma 20 (Lemma 10 Restated) *Suppose m satisfies the conditions in Theorem 5. With probability at least $1 - \delta$ over the initialization, there exists $\boldsymbol{\theta}' \in B(\boldsymbol{\theta}_0, \tilde{\Omega}(T^{3/2}))$, such that*

$$\begin{aligned}
 \mathbb{E}[\Psi(\boldsymbol{\theta}_0^2, \tilde{\Omega}(T^{3/2}))] &\leq \sum_{t=1}^{Tn} \mathbb{E}[(r_t - f(\mathbf{x}_t; \boldsymbol{\theta}'))^2/2] \\
 &\leq \mathcal{O} \left(\sqrt{\tilde{d} \log(1 + Tn) - 2 \log \delta + S + 1} \right)^2 \cdot \tilde{d} \log(1 + Tn).
 \end{aligned} \tag{C.1}$$

Proof

$$\begin{aligned}
 \mathbb{E} \left[\sum_{t=1}^{Tn} (r_t - f(\mathbf{x}_t; \boldsymbol{\theta}'))^2 \right] &= \sum_{t=1}^{Tn} (h(\mathbf{x}_t) - f(\mathbf{x}_t; \boldsymbol{\theta}'))^2 \\
 &\stackrel{(a)}{\leq} \mathcal{O} \left(\sqrt{\log \left(\frac{\det(\mathbf{A}_T)}{\det(\mathbf{I})} \right) - 2 \log \delta + S + 1} \right)^2 \sum_{t=1}^{Tn} \|g(\mathbf{x}_t; \boldsymbol{\theta}_0)\|_{\mathbf{A}_T^{-1}}^2 + 2Tn \cdot \mathcal{O} \left(\frac{T^2 L^3 \sqrt{\log m}}{m^{1/3}} \right) \\
 &\stackrel{(b)}{\leq} \mathcal{O} \left(\sqrt{\tilde{d} \log(1 + Tn) - 2 \log \delta + S + 1} \right)^2 \cdot (\tilde{d} \log(1 + Tn) + 1) + \mathcal{O}(1),
 \end{aligned}$$

where (a) is based on Lemma 22 and (b) is an application of Lemma 11 in Abbasi-Yadkori et al. (2011) and Lemma 26, and $\mathcal{O}(1)$ is induced by the choice of m . By ignoring $\mathcal{O}(1)$, The proof is completed. \blacksquare

Definition 21 Given the context vectors $\{\mathbf{x}_{\tau,\hat{i}}\}_{\tau=1}^T$ and the rewards $\{r_{\tau,\hat{i}}\}_{\tau=1}^T$, then we define the estimation $\hat{\boldsymbol{\theta}}_t$ via ridge regression:

$$\begin{aligned}\mathbf{A}_t &= \mathbf{I} + \sum_{\tau=1}^t g(\mathbf{x}_{\tau,\hat{i}}; \boldsymbol{\theta}_0) g(\mathbf{x}_{\tau,\hat{i}}; \boldsymbol{\theta}_0)^\top \\ \mathbf{b}_t &= \sum_{\tau=1}^t r_{\tau,\hat{i}} g(\mathbf{x}_{\tau,\hat{i}}; \boldsymbol{\theta}_0) \\ \hat{\boldsymbol{\theta}}_t &= \mathbf{A}_t^{-1} \mathbf{b}_t\end{aligned}$$

Lemma 22 Suppose m satisfies the conditions in Theorem 5. With probability at least $1 - \delta$ over the initialization, there exists $\boldsymbol{\theta}' \in B(\boldsymbol{\theta}_0, \tilde{\Omega}(T^{3/2}))$ for all $t \in [T]$, such that

$$\begin{aligned}& |h(\mathbf{x}_t) - f(\mathbf{x}_t; \boldsymbol{\theta}')| \\ & \leq \mathcal{O} \left(\sqrt{\log \left(\frac{\det(\mathbf{A}_t)}{\det(\mathbf{I})} \right) - 2 \log \delta + S + 1} \right) \|g(\mathbf{x}_t; \boldsymbol{\theta}_0)\|_{\mathbf{A}_t^{-1}} + \mathcal{O} \left(\frac{T^2 L^3 \sqrt{\log m}}{m^{1/3}} \right)\end{aligned}\quad (\text{C.2})$$

Proof Given a set of context vectors $\{\mathbf{x}_t\}_{t=1}^{Tn}$ with the ground-truth function h and a fully-connected neural network f , we have

$$\begin{aligned}& |h(\mathbf{x}_t) - f(\mathbf{x}_t; \boldsymbol{\theta}')| \\ & \leq \left| h(\mathbf{x}_t) - \langle g(\mathbf{x}_t; \boldsymbol{\theta}_0), \hat{\boldsymbol{\theta}}_t \rangle \right| + \left| f(\mathbf{x}_t; \boldsymbol{\theta}') - \langle g(\mathbf{x}_t; \boldsymbol{\theta}_0), \hat{\boldsymbol{\theta}}_t \rangle \right|\end{aligned}$$

where $\boldsymbol{\theta}'$ is the estimation of ridge regression from Definition 21. Then, based on the Lemma 24, there exists $\boldsymbol{\theta}^* \in \mathbf{R}^P$ such that $h(\mathbf{x}_t) = \langle g(\mathbf{x}_t, \boldsymbol{\theta}_0), \boldsymbol{\theta}^* \rangle$. Thus, we have

$$\begin{aligned}& \left| h(\mathbf{x}_t) - \langle g(\mathbf{x}_t; \boldsymbol{\theta}_0), \hat{\boldsymbol{\theta}}_t \rangle \right| \\ & = \left| \langle g(\mathbf{x}_t, \boldsymbol{\theta}_0), \boldsymbol{\theta}^* \rangle - \langle g(\mathbf{x}_t, \boldsymbol{\theta}_0), \hat{\boldsymbol{\theta}}_t \rangle \right| \\ & \leq \mathcal{O} \left(\sqrt{\log \left(\frac{\det(\mathbf{A}_t)}{\det(\mathbf{I})} \right) - 2 \log \delta + S} \right) \|g(\mathbf{x}_t; \boldsymbol{\theta}_0)\|_{\mathbf{A}_t^{-1}}\end{aligned}$$

where the final inequality is based on the Theorem 2 in Abbasi-Yadkori et al. (2011), with probability at least $1 - \delta$, for any $t \in [T]$.

Second, we need to bound

$$\begin{aligned}& \left| f(\mathbf{x}_t; \boldsymbol{\theta}') - \langle g(\mathbf{x}_t; \boldsymbol{\theta}_0), \hat{\boldsymbol{\theta}}_t \rangle \right| \\ & \leq \left| f(\mathbf{x}_t; \boldsymbol{\theta}') - \langle g(\mathbf{x}_t; \boldsymbol{\theta}_0), \boldsymbol{\theta}' - \boldsymbol{\theta}_0 \rangle \right| \\ & \quad + \left| \langle g(\mathbf{x}_t; \boldsymbol{\theta}_0), \boldsymbol{\theta}' - \boldsymbol{\theta}_0 \rangle - \langle g(\mathbf{x}_t; \boldsymbol{\theta}_0), \hat{\boldsymbol{\theta}}_t \rangle \right|\end{aligned}$$

To bound the above inequality, we first bound

$$\begin{aligned} & |f(\mathbf{x}_t; \boldsymbol{\theta}') - \langle g(\mathbf{x}_t; \boldsymbol{\theta}_0), \boldsymbol{\theta}' - \boldsymbol{\theta}_0 \rangle| \\ &= |f(\mathbf{x}_t; \boldsymbol{\theta}') - f(\mathbf{x}_t; \boldsymbol{\theta}_0) - \langle g(\mathbf{x}_t; \boldsymbol{\theta}_0), \boldsymbol{\theta}' - \boldsymbol{\theta}_0 \rangle| \\ &\leq \mathcal{O}(\omega^{4/3} L^3 \sqrt{\log m}) \end{aligned}$$

where we initialize $f(\mathbf{x}_t; \boldsymbol{\theta}_0) = 0$ following (Zhou et al., 2020) and the inequality is derived by Lemma 13 with $\omega = \frac{\mathcal{O}(t^{3/2})}{m^{1/4}}$. Next, we need to bound

$$\begin{aligned} & |\langle g(\mathbf{x}_t; \boldsymbol{\theta}_0), \boldsymbol{\theta}' - \boldsymbol{\theta}_0 \rangle - \langle g(\mathbf{x}_t; \boldsymbol{\theta}_0), \widehat{\boldsymbol{\theta}}_t \rangle| \\ &= |\langle g(\mathbf{x}_t; \boldsymbol{\theta}_0), (\boldsymbol{\theta}' - \boldsymbol{\theta}_0 - \widehat{\boldsymbol{\theta}}_t) \rangle| \\ &\leq \|g(\mathbf{x}_t; \boldsymbol{\theta}_0)\|_{\mathbf{A}_t^{-1}} \cdot \|\boldsymbol{\theta}' - \boldsymbol{\theta}_0 - \widehat{\boldsymbol{\theta}}_t\|_{\mathbf{A}_t} \\ &\leq \|g(\mathbf{x}_t; \boldsymbol{\theta}_0)\|_{\mathbf{A}_t^{-1}} \cdot \|\mathbf{A}_t\|_2 \cdot \|\boldsymbol{\theta}' - \boldsymbol{\theta}_0 - \widehat{\boldsymbol{\theta}}_t\|_2. \end{aligned}$$

Due to the Lemma 26 and Lemma 25, we have

$$\|\mathbf{A}_t\|_2 \cdot \|\boldsymbol{\theta}' - \boldsymbol{\theta}_0 - \widehat{\boldsymbol{\theta}}_t\|_2 \leq (1 + t\mathcal{O}(L)) \cdot \frac{1}{1 + \mathcal{O}(tL)} = \mathcal{O}(1).$$

Finally, putting everything together, the proof is completed. \blacksquare

Definition 23

$$\begin{aligned} \mathbf{G}^{(0)} &= [g(\mathbf{x}_{1,i}; \boldsymbol{\theta}_0), \dots, g(\mathbf{x}_{T,i}; \boldsymbol{\theta}_0)] \in \mathbb{R}^{p \times T} \\ \mathbf{G}_0 &= [g(\mathbf{x}_1; \boldsymbol{\theta}_0), \dots, g(\mathbf{x}_{Tn}; \boldsymbol{\theta}_0)] \in \mathbb{R}^{p \times Tn} \\ \mathbf{r} &= (r_{1,i}, \dots, r_{T,i}) \in \mathbb{R}^T \end{aligned}$$

$\mathbf{G}^{(0)}$ and \mathbf{r} are formed by the selected contexts and observed rewards in T rounds, \mathbf{G}_0 are formed by all the presented contexts.

Inspired by Lemma B.2 in (Zhou et al., 2020), with $\eta = m^{-1/4}$ we define the auxiliary sequence following :

$$\boldsymbol{\theta}_0 = \boldsymbol{\theta}^{(0)}, \quad \boldsymbol{\theta}^{(j+1)} = \boldsymbol{\theta}^{(j)} - \eta \left[\mathbf{G}^{(0)} \left([\mathbf{G}^{(0)}]^\top (\boldsymbol{\theta}^{(j)} - \boldsymbol{\theta}_0) - \mathbf{r} \right) + \lambda (\boldsymbol{\theta}^{(j)} - \boldsymbol{\theta}_0) \right]$$

Lemma 24 Suppose m satisfies the conditions in Theorem 5. With probability at least $1 - \delta$ over the initialization, for any $t \in [T], i \in [K]$, the result uniformly holds:

$$h(\mathbf{x}_{t,i}) = \langle g(\mathbf{x}_{t,i}; \boldsymbol{\theta}_0), \boldsymbol{\theta}^* - \boldsymbol{\theta}_0 \rangle.$$

Proof Based on Lemma 27 with proper choice of ϵ , we have

$$\mathbf{G}_0^\top \mathbf{G}_0 \succeq \mathbf{H} - \|\mathbf{G}_0^\top \mathbf{G}_0 - \mathbf{H}\|_F \mathbf{I} \succeq \mathbf{H} - \lambda_0 \mathbf{I} / 2 \succeq \mathbf{H} / 2 \succeq 0.$$

Define $\mathbf{h} = [h(\mathbf{x}_1), \dots, h(\mathbf{x}_{Tn})]$. Suppose the singular value decomposition of \mathbf{G}_0 is $\mathbf{P}\mathbf{A}\mathbf{Q}^\top$, $\mathbf{P} \in \mathbb{R}^{p \times Tn}$, $\mathbf{A} \in \mathbb{R}^{Tn \times Tn}$, $\mathbf{Q} \in \mathbb{R}^{Tn \times Tn}$, then, $\mathbf{A} \succeq 0$. Define $\boldsymbol{\theta}^* = \boldsymbol{\theta}_0 + \mathbf{P}\mathbf{A}^{-1}\mathbf{Q}^\top \mathbf{h}$. Then, we have

$$\mathbf{G}_0^\top (\boldsymbol{\theta}^* - \boldsymbol{\theta}_0) = \mathbf{Q}\mathbf{A}\mathbf{P}^\top \mathbf{P}\mathbf{A}^{-1}\mathbf{Q}^\top \mathbf{h} = \mathbf{h}.$$

which leads to

$$\sum_{t=1}^{Tn} (h(\mathbf{x}_t) - \langle g(\mathbf{x}_t; \boldsymbol{\theta}_0), \boldsymbol{\theta}^* - \boldsymbol{\theta}_0 \rangle) = 0.$$

Therefore, the result holds:

$$\|\boldsymbol{\theta}^* - \boldsymbol{\theta}_0\|_2^2 = \mathbf{h}^\top \mathbf{Q} \mathbf{A}^{-2} \mathbf{Q}^\top \mathbf{h} = \mathbf{h}^\top (\mathbf{G}_0^\top \mathbf{G}_0)^{-1} \mathbf{h} \leq 2 \mathbf{h}^\top \mathbf{H}^{-1} \mathbf{h} \quad (\text{C.3})$$

■

Lemma 25 *There exist $\boldsymbol{\theta}' \in B(\boldsymbol{\theta}_0, \tilde{\mathcal{O}}(T^{3/2}L + \sqrt{T}))$, such that, with probability at least $1 - \delta$, the results hold:*

$$\begin{aligned} (1) \|\boldsymbol{\theta}' - \boldsymbol{\theta}_0\|_2 &\leq \frac{\tilde{\mathcal{O}}(T^{3/2}L + \sqrt{T})}{m^{1/4}} \\ (2) \|\boldsymbol{\theta}' - \boldsymbol{\theta}_0 - \hat{\boldsymbol{\theta}}_t\|_2 &\leq \frac{1}{1 + \mathcal{O}(TL)} \end{aligned}$$

Proof The sequence of $\boldsymbol{\theta}^{(j)}$ is updated by using gradient descent on the loss function:

$$\min_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}) = \frac{1}{2} \|[\mathbf{G}^{(0)}]^\top (\boldsymbol{\theta} - \boldsymbol{\theta}^{(0)}) - \mathbf{r}\|_2^2 + \frac{\lambda}{2} \|\boldsymbol{\theta} - \boldsymbol{\theta}^{(0)}\|_2^2.$$

For any $j > 0$, the results hold:

$$\|\mathbf{G}^{(0)}\|_F \leq \sqrt{T} \max_{t \in [T]} \|g(\mathbf{x}_t; \boldsymbol{\theta}_0)\|_2 \leq \mathcal{O}(\sqrt{TL}),$$

where the last inequality is held by Lemma 12. Finally, given the $j > 0$,

$$\|\boldsymbol{\theta}^{(j)} - \boldsymbol{\theta}^{(0)}\|_2 \leq \sum_{i=1}^j \eta \left[\mathbf{G}^{(0)} \left([\mathbf{G}^{(0)}]^\top (\boldsymbol{\theta}^{(i)} - \boldsymbol{\theta}_0) - \mathbf{r} \right) + \lambda (\boldsymbol{\theta}^{(i)} - \boldsymbol{\theta}_0) \right] \leq \frac{\mathcal{O}(j(TL\sqrt{T/\lambda} + \sqrt{T\lambda}))}{m^{1/4}}. \quad (\text{C.4})$$

For (2), by standard results of gradient descent on ridge regression, $\boldsymbol{\theta}^{(j)}$, and the optimum is $\boldsymbol{\theta}^{(0)} + \hat{\boldsymbol{\theta}}_t$. Therefore, we have

$$\begin{aligned} \|\boldsymbol{\theta}^{(j)} - \boldsymbol{\theta}^{(0)} - \hat{\boldsymbol{\theta}}_t\|_2^2 &\leq [1 - \eta\lambda]^j \frac{2}{\lambda} \left(\mathcal{L}(\boldsymbol{\theta}^{(0)}) - \mathcal{L}(\boldsymbol{\theta}^{(0)} + \hat{\boldsymbol{\theta}}_t) \right) \\ &\leq \frac{2(1 - \eta\lambda)^j}{\lambda} \mathcal{L}(\boldsymbol{\theta}^{(0)}) \\ &= \frac{2(1 - \eta m \lambda)^j}{\lambda} \frac{\|\mathbf{r}\|_2^2}{2} \\ &\leq \frac{T(1 - \eta\lambda)^j}{\lambda}. \end{aligned}$$

By setting $\lambda = 1$ and $j = \log((T + \mathcal{O}(T^2L))^{-1}) / \log(1 - m^{-1/4})$, we have $\|\boldsymbol{\theta}^{(j)} - \boldsymbol{\theta}_0 - \hat{\boldsymbol{\theta}}_t\|_2^2 \leq \frac{1}{1 + \mathcal{O}(TL)}$. Replacing k and λ in (C.4) finishes the proof. ■

Lemma 26 *Suppose m satisfies the conditions in Theorem 5. With probability at least $1 - \delta$ over the initialization, the result holds:*

$$\begin{aligned} \|\mathbf{A}_T\|_2 &\leq 1 + \mathcal{O}(TL), \\ \log \frac{\det \mathbf{A}_T}{\det \mathbf{I}} &\leq \tilde{d} \log(1 + Tn) + 1. \end{aligned}$$

Proof Based on the Lemma 12, for any $t \in [T]$, $\|g(\mathbf{x}_t; \boldsymbol{\theta}_0)\|_2 \leq \mathcal{O}(\sqrt{L})$. Then, for the first item:

$$\begin{aligned} \|\mathbf{A}_T\|_2 &= \|\mathbf{I} + \sum_{t=1}^T g(\mathbf{x}_{t,\hat{i}}; \boldsymbol{\theta}_0)g(\mathbf{x}_{t,\hat{i}}; \boldsymbol{\theta}_0)^\top\|_2 \\ &\leq \|\mathbf{I}\|_2 + \left\| \sum_{t=1}^T g(\mathbf{x}_{t,\hat{i}}; \boldsymbol{\theta}_0)g(\mathbf{x}_{t,\hat{i}}; \boldsymbol{\theta}_0)^\top \right\|_2 \\ &\leq 1 + \sum_{t=1}^T \|g(\mathbf{x}_{t,\hat{i}}; \boldsymbol{\theta}_0)\|_2^2 \leq 1 + \mathcal{O}(TL). \end{aligned}$$

Next, we have

$$\log \frac{\det(\mathbf{A}_T)}{\det(\mathbf{I})} \leq \log \det\left(\mathbf{I} + \sum_{t=1}^{Tn} g(\mathbf{x}_t; \boldsymbol{\theta}_0)g(\mathbf{x}_t; \boldsymbol{\theta}_0)^\top\right) = \det(\mathbf{I} + \mathbf{G}_0 \mathbf{G}_0^\top)$$

Then, we have

$$\begin{aligned} &\log \det(\mathbf{I} + \mathbf{G}_0 \mathbf{G}_0^\top) \\ &= \log \det(\mathbf{I} + \mathbf{H} + (\mathbf{G}_0 \mathbf{G}_0^\top - \mathbf{H})) \\ &\leq \log \det(\mathbf{I} + \mathbf{H}) + \langle (\mathbf{I} + \mathbf{H})^{-1}, (\mathbf{G}_0 \mathbf{G}_0^\top - \mathbf{H}) \rangle \\ &\leq \log \det(\mathbf{I} + \mathbf{H}) + \|(\mathbf{I} + \mathbf{H})^{-1}\|_F \|\mathbf{G}_0 \mathbf{G}_0^\top - \mathbf{H}\|_F \\ &\leq \log \det(\mathbf{I} + \mathbf{H}) + \sqrt{T} \|\mathbf{G}_0 \mathbf{G}_0^\top - \mathbf{H}\|_F \\ &\leq \log \det(\mathbf{I} + \mathbf{H}) + 1 \\ &= \tilde{d} \log(1 + Tn) + 1. \end{aligned}$$

The first inequality is because the concavity of $\log \det$; The third inequality is due to $\|(\mathbf{I} + \mathbf{H})^{-1}\|_F \leq \|\mathbf{I}^{-1}\|_F \leq \sqrt{T}$; The last inequality is because of the choice the m , based on Lemma 27; The last equality is because of the Definition of \tilde{d} . The proof is completed. ■

Lemma 27 *For any $\delta \in (0, 1)$, if $m = \Omega\left(\frac{L^6 \log(TnL/\delta)}{(\epsilon/Tn)^4}\right)$, then with probability at least $1 - \delta$, the results hold:*

$$\|\mathbf{G}_0 \mathbf{G}_0^\top - \mathbf{H}\|_F \leq \epsilon.$$

Proof This is an application of Lemma B.1 in (Zhou et al., 2020) by properly setting ϵ . ■