NAORL: Network Feature Aware Offline Reinforcement Learning for Real Time Bandwidth Estimation

Wei Zhang
Bytedance
Shenzhen, Guangdong, China
zhangwei.666@bytedance.com

Xuefeng Tao
Bytedance
Shenzhen, Guangdong, China
taoxuefeng@bytedance.com

Jianan Wang
Bytedance
Hangzhou, Zhejiang, China
wangjianan.324@bytedance.com

ABSTRACT

Bandwidth Estimation(BWE) is the most important and challenging problem for Real Time Communication(RTC) systems. The rule-based BWE is designed with hand-crafted rules, which mainly depend on human knowledge, and therefore is difficult to generalize to unknown scenarios. Learning-based BWE algorithms, especially online reinforcement learning-based algorithms, are proposed to explore new decisions in complex network environments adaptively. However, these algorithms require frequent interactions with the environment, which would cause catastrophic experience for RTC users.

In this paper, we propose NAORL, a Network feature Aware Offline Reinforcement Learning for BWE in RTC applications. First, we devise a network-aware feature pre-training scheme, which extracts network-aware feature representation from both emulated and testbed data. Notably, pre-training is conducted in a self-regressive manner, where no labeled data is required, and its performance would be increased with richer and more diverse data. Second, we adopt an effective offline reinforcement learning algorithm IQL to learn experiences from existing data. Third, we design a multi-expert module to enhance model robustness. Furthermore, we devise a method of curriculum learning to facilitate efficient model learning. Finally, we design a set of metrics to evaluate the accuracy of NAORL and conduct extensive ablation experiments. The evaluation results demonstrate that all building blocks in NAORL improve accuracy significantly. In addition, NAORL achieves superior accuracy compared to the mmsys baseline model. The source code can be found at *https://github.com/bytedance/offline-RL-congestion-control*.

ACM Reference Format:
Wei Zhang, Xuefeng Tao, and Jianan Wang. 2024. NAORL: Network Feature Aware Offline Reinforcement Learning for Real Time Bandwidth Estimation. In *ACM Multimedia Systems Conference 2024 (MMSys '24), April 15–18, 2024, Bari, Italy*. ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3625468.3652177

INTRODUCTION

Real-Time Communication(RTC) has attracted significant interests from both industry and academics. There is an increasing demand for RTC based applications, such as video conferencing, online education, and chatting rooms.

Bandwidth Estimation (BWE) is the most important and challenging issue in real-time communication (RTC) applications. Existing BWE methods could be divided into two categories, i.e., classical rule-based BWE, and learning-based BWE. The rule-based BWE [3], [4], [12], [17], [2] is designed with hand-crafted rules, which mainly depend on human knowledge, and therefore is difficult to generalize to unknown scenarios. Learning-based BWE algorithms [1], [19], [18], especially online reinforcement learning based algorithms, are proposed to explore new decisions in complex network environments adaptively [1], [19], and [18]. However, these algorithms require frequent interactions with the environment, which would cause catastrophic experience for RTC users.

Different from online reinforcement learning, the offline reinforcement learning algorithm [16] [6] does not require interaction with the environment, eliminating the possibility of catastrophic events occurring during model optimization. By observing and learning from a large amount of data generated by diverse BWE algorithms, offline reinforcement can learn an even better BWE algorithm.

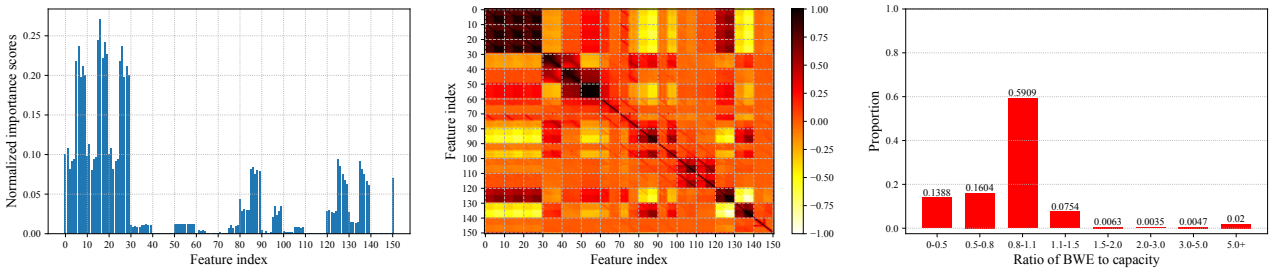
However, learning from offline data is not trivial. There are three challenging problems with offline reinforcement learning. First, how to ensure the generalization capability and robustness of the model when it is trained offline. Second, how to deal with the distribution shift problem if the distribution between the collected training data and online data is inconsistent. Third, how to solve the compounding error, since the output of one action would influence a series of subsequent actions.

In this paper, we propose NAORL, a Network feature Aware Offline Reinforcement Learning for BWE in RTC applications. First, we devise a network-aware feature pre-training scheme, which extracts network-aware feature representation from both emulated and testbed data. Notably, pre-training is conducted in a self-regressive manner, where no labeled data is required, and its performance would be increased with richer and more diverse data. Second, we adopt an effective offline reinforcement learning algorithm, i.e., implicit q-learning algorithm, to learn experiences from existing data. Third, we design a multi-expert module to enhance model robustness. Furthermore, we devise a method of curriculum learning to better guide model learning. Finally, we design a set of metrics to evaluate the accuracy of NAORL and conduct extensive ablation experiments. The evaluation results demonstrate that NAORL achieves better accuracy than the mmsys baseline model.

Contributions

The contributions of this paper are summarized as follows.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
MMSys'22, April 15-18, 2024, Bari, Italy
© 2024 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 979-8-4007-0412-3/24/04
https://doi.org/10.1145/3625468.3652177



(a) Feature Importance **(b) Feature Correlation Heatmap** **(c) Accuracy of Policy Behaviors**

Figure 1: Data Analysis

- We propose NAORL, a Network feature Aware Offline Reinforcement Learning algorithm for Bandwidth Estimation. The proposed scheme is trained and evaluated without any interaction with the environments, and can be optimized without causing any catastrophic experience problems for RTC users.
- We propose a network-aware feature pre-training scheme, which tries to extract network-aware features from the emulated and testbed environment. To the best of knowledge, this is the first work that tries to extract network-aware features in a self-regressive manner. We believe that with the enhancement of data diversity, these learned features can be widely applied to downstream tasks.
- We propose a series of building blocks to enhance the performance of NAORL, which includes curriculum learning, selecting samples with acceptable prediction accuracy, and multi-expert decisions. Each building block significantly contributes to improving prediction accuracy.
- We design a set of metrics to evaluate the accuracy of NAORL, and conduct extensive ablation experiments. The evaluation results demonstrate that NAORL achieves better accuracy than the mmsys baseline model.

PRELIMINARIES

Offline Reinforcement Learning(ORL) is proposed to perform reinforcement learning based on previously collected data without any interaction with the environment [14]. The ORL aims at learning a better policy over behavior policies, while minimizing deviations from behavior policies. Most existing ORL algorithms require querying the value of out-of-sample actions, which would cause distribution shift and performance degradation. Implicit Q-Learning (IQL) [10] proposes approximating the value of best actions with expectile regression, without querying any unseen samples, which achieves state-of-the-art performance. Due to the excellent performance and properties provided by IQL, we leverage IQL to train our model.

IQL is composed of four networks, i.e, value network, Q-network, target Q-network, and policy network. The value network is updated with $\psi \leftarrow \psi - \lambda_V \nabla_\psi L_V(\psi)$ where

$$L_V(\psi) = \mathbb{E}_{(s,a)\sim\mathcal{D}}[L_2^\tau(Q_{\hat{\theta}}(s,a) - V_\psi(s))]. \quad (1)$$

The Q-network is updated with $\theta \leftarrow \theta - \lambda_Q \nabla_\theta L_Q(\theta)$, where

$$L_Q(\theta) = \mathbb{E}_{(s,a,s\prime)\sim\mathcal{D}}[(r(s,a) + \gamma V_\psi(s\prime) - Q_\theta(s,a))^2]. \quad (2)$$

The target-Q network is updated with $\hat{\theta} \leftarrow (1-\alpha)\hat{\theta} + \alpha\theta$

Finally, the policy is extracted with advantage weighted regression $\phi \leftarrow \phi - \lambda_\pi \nabla_\phi L_\pi(\phi)$, where

$$L_\pi(\phi) = \mathbb{E}_{(s,a)\sim\mathcal{D}}[exp(\beta(Q_{\hat{\theta}(s,a)} - V_\psi(s)))log\pi_\phi(a|s)]. \quad (3)$$

DATA ANALYSIS

Before constructing models, it is necessary to carry out data analysis to have a comprehensive understanding of the provided data.

Feature analysis

We employ a linear regression methodology to assess the significance of network features in relation to audio and video objective QoE scores.

From Fig.1(a), it is evident that the objective score predominantly depends on a series of metrics, including the long&short term receiving rate(0-9), number of received packets(10-19), received bytes(20-29), long&short term packet interarrival time(80-89), long&short term video packets probability(120-129), and audio packets probability(130-139).

The correlation among network features is calculated using Pearson correlation. As shown in Fig.1(b), the correlation among network features becomes quite explicit. It's worth noting that, the receiving rate exhibits a substantial correlation with features that are closely tied to the objective score.

From this feature analysis, it can be inferred that, the audio and video objective QoE scores are significantly influenced by only several network features, and these features are all highly correlated to the receiving rate feature. Since receiving rate is mainly affected by bandwidth prediction, we postulate that improving the accuracy of bandwidth predictions will lead to elevated objective QoE scores.

Accuracy of Policy Behaviors

Fig.1(c) demonstrates the precision of policy behaviors in the emulated dataset [8]. We have the following observations: 1. Almost 30% of the samples show an underestimation. 2. More than 10% of the samples present an overestimation. 3. Only 59.09% of the samples demonstrate a satisfactory accuracy. In particular, over 13.88% of the samples exhibit bandwidth estimation below half of

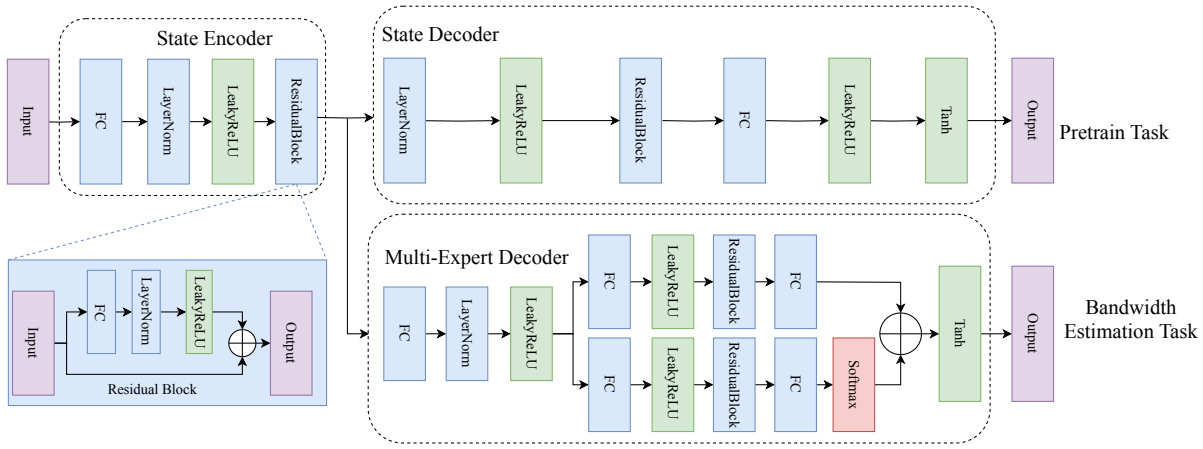


Figure 2: NAORL Network Architecture

the ground truth, and more than 10% of the samples output bandwidth estimation surpassing 1.1 times of the ground truth. The above observations motivate us to pay more attention to the noise within the dataset.

BUILDING BLOCKS FOR NAORL

The architecture of NAORL is demonstrated in Fig.2. NAORL is composed of a pre-training task, and a bandwidth estimation task. The pre-training task is designed to extract network-aware features, which is expected to enhance the performance of NAORL. The bandwidth estimation task extracts network-aware features with the pretrained state encoder, and outputs the bandwidth estimation with a multi-expert decoder, which imitates outputting bandwidth estimation from multiple-experts. In this section, we aim to detail building blocks of NAORL.

pre-training on Diverse Data

A lot of previous reinforcement learning based algorithms aim to map the input to a reward, where the network feature representation is ignored. Different from prior work, in this paper, we propose to pretrain network features on diverse datasets, so that we could discover a more efficient representation of network features. In this paper, the pre-training task is designed to predict the next observation given the current observation, with the objective of minimizing the mean square error of the two observations. Note that, the pretrain is designed to be trained in a self-supervised manner, where no labed data is required. We believe that with more and richer data added to the training, the generalization capability of the pretrained model will be strengthened.

As shown in Fig.2, the pre-training task is performed with a state encoder and a state decoder. The state encoder is designed to extract network-aware features, while the state decoder is devised to perform the generation task.

Multi-expert Decision

To strengthen the accuracy and robustness of the prediction result, we design a multi-expert module. Specifically, instead of outputing a prediction from a single expert, we simulate multiple experts

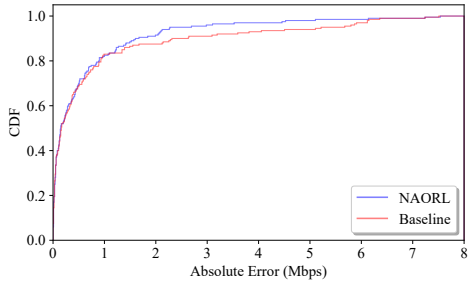


Figure 3: Model Prediction Error

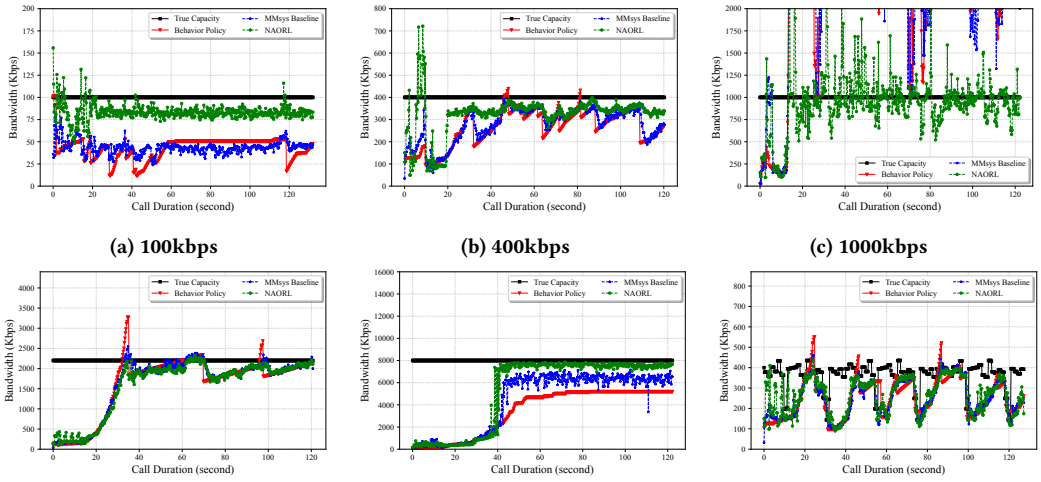
in the network structure as shown in the Fig.2. The multi-expert decoder will output a weighted sum of predictions coming from different experts.

Curriculum Learning

To ensure the offline agent learns efficiently, we propose to train the agent with curriculum learning. Specifically, we first input data whose ground truth bandwidth falls in the range of 0-100Kbps. After training for a few epochs, we input data with a larger range of 0-300Kbps. The remaining data is trained gradually in a similar way. Note that, the number of iterations for different samples is proportional to the number of samples, so that underfit or overfit will not occur.

Selecting Samples With Acceptable Prediction Accuracy

As illustrated in Fig.1(c), the training dataset contains considerable inaccurate data, where the estimated bandwidth significantly deviates from the ground truth. Therefore, we propose filtering out inaccurate data, and focusing solely on predictions exhibiting high precision. Specifically, when the ratio between the prediction and the ground truth falls outside the range 0.8-1.1, the corresponding samples will not be utilized.



(a) 100kbps (b) 400kbps (c) 1000kbps

(d) 2200kbps (e) 8000kbps (f) 100-500kbps

Figure 4: Bandwidth Estimation Over Different Bandwidth Limitations

TRAIN NAORL WITH IQL

In this section, we present the details of training NAORL.

State

The State is 150-dimensional data, which is derived from 15 different network statistics over 5 short and 5 long term monitoring intervals. Specifically, the 15 network features are Receiving rate, Number of received packets, Received bytes, Queuing delay, Delay, Minimum seen delay, Delay ratio, Delay average minimum difference, Packet interarrival time, Packet jitter, Packet loss ratio, Average number of lost packets, Video packets probability, Audio packets probability, Probing packets probability. The detailed information of these network features can be found in [13].

Action

The action for offline reinforcement learning is the predicted bandwidth, ranging from 10kbps to 8Mbps.

Reward

The Reward for the IQL is mainly based on the distance between the predicted bandwidth b_t and the ground truth c_t. The reward is computed with the following formula.

$$reward = \begin{cases} \alpha e^{-\beta_1 (r_t - 1)^2} & \text{if } r_t > 1.0 \\ \alpha e^{-\beta_2 (r_t - 1)^2} & \text{if } 0.8 <= r_t <= 1.0 \\ \alpha e^{-\beta_3 (r_t - 1)^2} & \text{if } r_t < 0.8 \end{cases} \quad (4)$$

where $r_t = b_t / c_t$, $\alpha = 10$, $\beta_1 = 80$, $\beta_2 = 8$, $\beta_3 = 40$. As illustrated in Equation 4, overestimation has the lowest reward since it would cause ongoing and serious congestion, underestimation possesses a relatively greater reward than overestimation, while estimation close to the ground truth is assigned with the highest reward.

Finally, NAORL is trained with IQL as explained in the Preliminaries.

EVALUATION

In this section, we first illustrate the metrics used for guiding optimization. Then we make a thorough comparison on the bandwidth prediction between NAORL and other algorithms. Finally, we present the results of ablation experiments.

Metrics Used for Guiding Optimization

A fundamental challenge for offline reinforcement learning is that it is difficult to evaluate the model without online interaction. To solve this problem, we design a metric that will serve as an efficient tool for optimization. Specifically, we first compute the ratio between the predicted bandwidth and the ground truth. Subsequently, the ratio is divided into multiple intervals. We define accuracy as the proportion of ratios falling within the range 0.8-1.1, underestimation as the proportion of ratios below 0.8, and overestimation as the proportion of ratios larger than 1.1. We observe that the result evaluated by this prediction accuracy is likely to be consistent with the online evaluation result.

Table 1: Model Prediction Accuracy

Model	*mse*	e^+	e^-
Baseline	3.844	0.3715	0.2121
NAORL	2.6125	0.0992	0.208

Model Prediction Accuracy

Fig. 3 demonstrates the Cumulative Distribution Function(CDF) of model prediction accuracy. As can be observed, NAORL achieves a relatively low prediction error. Table 1 concisely presents the statistics of model prediction error, where *mse* signifies the mean square error between prediction and ground truth, e^+ denotes the overestimation error rate, and e^- represents the underestimation error rate. Formulars for computation of *mse*, e^+, e^- can be found in [8]. From Table 1, it is evident that NAORL exhibits superior

Table 2: Comparison of Different Optimization Methods

ratio / name	0-0.5	0.5-0.8	0.8-1.1 accuracy	1.1-1.5	1.5-2.0	2.0-3.0	3.0-5.0	5.0-	>1.1 overestimation
mmsys baseline	0.1608	0.1522	0.5905	0.0474	0.0076	0.0066	0.0055	0.0293	0.0964
basic model	0.1473	0.173	0.4523	0.148	0.037	0.0093	0.0153	0.0178	0.2274
$ME - 10$	0.1199	0.1644	0.499	0.099	0.0331	0.0127	0.009	0.0349	0.1887
ME	0.1264	0.1528	0.5169	0.1076	0.0382	0.0159	0.009	0.0333	0.204
CL	0.1472	0.1466	0.5499	0.088	0.0179	0.012	0.0089	0.0294	0.1562
$CL + BA$	0.1398	0.126	0.6247	0.062	0.0194	0.0118	0.0099	0.0064	0.1095
$ME + BA$	0.1314	0.1197	0.6297	0.0692	0.0245	0.0161	0.0029	0.0064	0.1191
$ME + CL + BA$	0.1463	0.1327	0.6361	0.0539	0.0168	0.0055	0.0031	0.0056	0.0849
$PT + ME + CL + BA$	0.1521	0.116	**0.6787**	0.0254	0.0118	0.0073	0.0022	0.0065	**0.0532**

prediction accuracy. In addition, NAORL achieves significantly less overestimation, which could potentially lead to more severe QoE problems compared with underestimation.

Model Prediction Examples

Fig. 4 shows the predication accuracy of NAORL over different bandwidth limitation.

Fig. 4(a) and Fig. 4 (b) illustrate the performance of NAORL when the bandwidth limitation is low. As shown in both Fig. 4(a) and Fig. 4 (b), the bandwidth prediction of NAORL is close to the ground truth, while the behavior policy and mmsys baseline are relatively far away from the ground truth. Even under a bandwidth constraint of 100kbps, NAORL continues to forecast a more steady and precise value.

Fig.4(c) and Fig.4(d) present the performance of NAORL when there is a moderate bandwidth limitation. As shown in Fig.4(c), the prediction of NAORL is close to the ground truth, and seldom exceeds the ground truth with a large magnitude. In comparison, other algorithms exhibit a substantially higher incidence of outputting excessively high bandwidth. From Fig.4(d), it can be concluded that NAORL is not only more accurate, but also achieves better smoothness.

Fig.4(e) demonstrates NAORL's enhanced stability and accuracy.

Finally, Fig.4(f) illustrates the adaptability of different algorithms when the network fluctuates.

Ablation Study

In this paper, we have proposed several building blocks to improve model accuracy. In the ablation study, we show how building blocks influence model accuracy. In Table 2, the basic model is not optimized with any of our proposed building blocks. ME denotes the multiple expert module with 5 experts, $ME - 10$ means 10 experts are involved, CL represents the curriculum learning, BA means selecting best actions by filtering out inaccurate samples, and PT denotes the pre-training scheme. As shown in Table 1, all of the proposed building blocks improve accuracy significantly. The model with $PT + ME + CL + BA$ achieves the best performance on accuracy and overestimation.

LIMITATIONS AND DISCUSSIONS

Reward Function Selection

NAORL adopts the paradigm of pre-training on a large dataset (emulated dataset and testbed dataset) and fine-tuning on a small dataset (emulated dataset). During the fine-tuning process, the prediction accuracy, i.e., the difference between the prediction and the ground truth, serves as the reward function. These settings assume that more accurate predictions result in higher QoE scores, which may not necessarily be accurate as indicated in [8].

Furthermore, due to the absence of the ground truth in a real-world dataset, it is preferred to adopt a reward function based on QoE scores to steer the training.

We can enhance our training framework by selecting an appropriate QoE based reward function, so that both emulated and testbed datasets can be employed, with the aim of attaining the maximum QoE score.

Inaccurate Samples Filtering

Our empirical study concludes that filtering inaccurate samples improves bandwidth estimation accuracy on the emulated dataset. However, we do not advocate this approach for the following two considerations: Firstly, this methodology becomes impractical when the ground truth is unknown in the testbed dataset. Secondly, filtering inaccurate samples seems contradictory with the following two principles: 1. The IQL could learn and gain insights from both accurate and inaccurate samples. 2. Filtering an excessive number of samples would diminish sample efficiency, and potentially result in poor robustness and generalization. We intend to delve into the problem and find out the root cause behind it.

Feature Selection

Upon elaborate analysis of feature selection, it becomes evident that, the QoE score predominantly depends upon 60 features, where the other 90 features demonstrate minimal correlation with the objective QoE score. A potential approach would be to utilize the selected features instead of all noisy features as inputs, so that superior solutions could be discovered in a relatively smaller feature space, where the feature efficiency could also be augmented. In the future work, we aim to construct models based on selected features to improve model performance.

Pre-training Task Design

In this paper, we design a self-supervised pre-training paradigm to extract network-aware features. Concretely, the pre-training task is designed to predict the subsequent network observation based on the current network observation, with the objective of minimizing the mean square error of the two observations. This framework has the potential to expand subsequently. On the one hand, the pre-training model may be trained with a broader range of data without expensive labelling. On the other hand, we can devise novel pre-training tasks to acquire both network-aware and task-aware features.

RELATED WORK

Offline Reinforcement Learning

Offline Reinforcement Learning(ORL) is proposed to perform reinforcement learning without any interaction with the environment. The ORL can be classified into Imitation Learning based algorithms and the Reinforcement Learning based algorithms [14]. The imitation learning based algorithms apply a supervised paradigm to perform training, with the goal of minimizing the imitation error between the learned policy and the expert policy [7]. The reinforcement learning based algorithms aim at learning a better policy than behavior policies, while minimizing deviations from behavior policies, including action-constrained algorithms [5], [15], and value-regularized algorithms [11], [9].

Implicit Q-Learning [10] proposes approximating the value of best actions with expectile regression, without querying any unseen samples, which achieves SOTA performance. In this paper, due to the excellent performance and properties provided by IQL, we leverage IQL to train our model.

Bandwidth Estimation for RTC applications

Bandwidth Estimation is a highly important and challenging research topic, which wins great interests from both academics and industry. Existing methods can be divided into classical rule-based congestion control [3], [4], [12], [17], [2], and learning-based congestion control [1], [19], [18]. Yu et al. [16] collects data from heuristic algorithms, and adopt offline reinforcement learning algorithms to obtain a robust bandwidth estimation. Gottipati et al. [6] proposed an imitation learning based BWE algorithm for RTC applications.

In this paper, different from prior work, we propose NAORL, which adopts IQL and integrates a variety of key designs to obtain a better policy without introducing the distribution shift problem.

CONCLUSION

In this paper, we propose NAORL, a Network feature Aware Offline Reinforcement Learning for BWE in RTC applications. First, we devise a network-aware feature pre-training scheme, which extracts network-aware feature representation from emulated and testbed data. Notably, pre-training is conducted in a self-regressive manner, where no labeled data is required, and its performance can be augmented with richer and more diverse data. Second, we adopt an effective offline reinforcement learning algorithm IQL to learn experiences from existing data. Third, we design a multi-expert module to enhance model robustness. Furthermore, we devise a method of curriculum learning to facilitate efficient model learning. Finally, we design a set of metrics to evaluate the accuracy of NAORL and conduct extensive ablation experiments. The evaluation results demonstrate that all building blocks in NAORL improve accuracy significantly. In addition, NAORL achieves superior accuracy compared to the mmsys baseline model.

REFERENCES

[1] Soheil Abbasloo, Chen-Yu Yen, and H Jonathan Chao. 2020. Classic meets modern: A pragmatic learning-based congestion control for the internet. In *Proceedings of the Annual conference of the ACM Special Interest Group on Data Communication on the applications, technologies, architectures, and protocols for computer communication*. 632–647.

[2] Venkat Arun and Hari Balakrishnan. 2018. Copa: Practical {Delay-Based} congestion control for the internet. In *15th USENIX Symposium on Networked Systems Design and Implementation (NSDI 18)*. 329–342.

[3] Neal Cardwell, Yuchung Cheng, C Stephen Gunn, Soheil Hassas Yeganeh, and Van Jacobson. 2017. BBR: Congestion-based congestion control. *Commun. ACM* 60, 2 (2017), 58–66.

[4] Gaetano Carlucci, Luca De Cicco, Stefan Holmer, and Saverio Mascolo. 2017. Congestion control for web real-time communication. *IEEE/ACM Transactions on Networking* 25, 5 (2017), 2629–2642.

[5] Scott Fujimoto, David Meger, and Doina Precup. 2019. Off-policy deep reinforcement learning without exploration. In *International conference on machine learning*. PMLR, 2052–2062.

[6] Aashish Gottipati, Sami Khairy, Gabriel Mittag, Vishak Gopal, and Ross Cutler. 2023. Real-time Bandwidth Estimation from Offline Expert Demonstrations. *arXiv preprint arXiv:2309.13481* (2023).

[7] Jonathan Ho and Stefano Ermon. 2016. Generative adversarial imitation learning. *Advances in neural information processing systems* 29 (2016).

[8] Sami Khairy, Gabriel Mittag, Scott Inglis, Vishak Gopal, Mehrsa Golestaneh, Ross Cutler, Francis Yan, and Zhixiong Niu. 2024. ACM MMSys 2024 Bandwidth Estimation in Real Time Communications Challenge. *arXiv preprint arXiv:2403.06324* (2024).

[9] Ilya Kostrikov, Rob Fergus, Jonathan Tompson, and Ofir Nachum. 2021. Offline reinforcement learning with fisher divergence critic regularization. In *International Conference on Machine Learning*. PMLR, 5774–5783.

[10] Ilya Kostrikov, Ashvin Nair, and Sergey Levine. 2021. Offline reinforcement learning with implicit q-learning. *arXiv preprint arXiv:2110.06169* (2021).

[11] Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. 2020. Conservative q-learning for offline reinforcement learning. *Advances in Neural Information Processing Systems* 33 (2020), 1179–1191.

[12] Tong Meng, Neta Rozen Schiff, P Brighten Godfrey, and Michael Schapira. 2020. PCC proteus: Scavenger transport and beyond. In *Proceedings of the Annual conference of the ACM Special Interest Group on Data Communication on the applications, technologies, architectures, and protocols for computer communication*. 615–631.

[13] Microsoft. 2023. 2nd Bandwidth Estimation Challenge at ACM MMSys 2024. https://www.microsoft.com/en-us/research/academic-program/bandwidth-estimation-challenge/data/. Accessed: (2024.1).

[14] Rafael Figueiredo Prudencio, Marcos ROA Maximo, and Esther Luna Colombini. 2023. A survey on offline reinforcement learning: Taxonomy, review, and open problems. *IEEE Transactions on Neural Networks and Learning Systems* (2023).

[15] Yifan Wu, George Tucker, and Ofir Nachum. 2019. Behavior regularized offline reinforcement learning. *arXiv preprint arXiv:1911.11361* (2019).

[16] Chen-Yu Yen, Soheil Abbasloo, and H Jonathan Chao. 2023. Computers Can Learn from the Heuristic Designs and Master Internet Congestion Control. In *Proceedings of the ACM SIGCOMM 2023 Conference*. 255–274.

[17] Yasir Zaki, Thomas Pötsch, Jay Chen, Lakshminarayanan Subramanian, and Carmelita Görg. 2015. Adaptive congestion control for unpredictable cellular networks. In *Proceedings of the 2015 ACM Conference on Special Interest Group on Data Communication*. 509–522.

[18] Huanhuan Zhang, Anfu Zhou, Yuhan Hu, Chaoyue Li, Guangping Wang, Xinyu Zhang, Huadong Ma, Leilei Wu, Aiyun Chen, and Changhui Wu. 2021. Loki: improving long tail performance of learning-based real-time video adaptation by fusing rule-based models. In *Proceedings of the 27th Annual International Conference on Mobile Computing and Networking*. 775–788.

[19] Huanhuan Zhang, Anfu Zhou, Jiamin Lu, Ruoxuan Ma, Yuhan Hu, Cong Li, Xinyu Zhang, Huadong Ma, and Xiaojiang Chen. 2020. OnRL: improving mobile video telephony via online reinforcement learning. In *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*. 1–14.