Multi-modal Contextual Bandit for Recommendation

Group7: Eungi Kim[1], Eunyi Lyou[1], Juneau Jung[2], Kwangeun Yeo[1]*
Graduate School of Data Science[1] Dept. of ECE[2]
Seoul National University
{kuman5262, onlyou0416, sean2ie, kwangeun.yeo}@snu.ac.kr

1 Problem Formulation

Recommendation systems have achieved widespread adoption in real-world applications. However, traditional collaborative filtering and content-based filtering methods rely on static models learned from training data and may not be suitable for dynamically changing environments with constantly changing users and items [1]. In order to address this issue, we propose a multi-modal contextual bandit algorithm that can adapt to dynamically changing environments, while also leveraging high-quality contextual information through multi-modal learning techniques. By combining these two approaches, we aim to develop a more effective and efficient recommendation system that can keep up with the constantly evolving nature of real-world applications.

2 Background and Related Work

Recommendation Systems Traditional recommendation systems have relied on non-Reinforcement Learning (RL) algorithms, which suffer from limitations such as poor generalization to new users and items, and inability to handle cold-start scenarios [1, 2]. Recent research has proposed RL-based Contextual Bandit Recommendation that can learn from user interactions and adapt to dynamic environments to personalize news article recommendations [3]. However, these approaches typically rely on single-modality inputs on textual features, which may limit their ability to capture the complexity of item attributes and user preferences. In this paper, we seek to address this limitation by incorporating multi-modal data, such as video and text, to enhance the quality of user and item representations.

Multi-modal Learning With the availability of large multi-modal datasets and powerful computational resources, multi-modal learning has become a popular approach for solving complex machine learning problems. By integrating information from various modalities, including vision, text, video, and audio, recent research has demonstrated significant improvements in downstream task performance. For instance, the CLIP, VATT, and VilBERT ([4, 5, 6]) models have shown innovative performance in tasks such as image-text retrieval, visual question answering, and multi-modal classification. These advancements highlight the potential for multi-modal contextual bandit models to leverage diverse sources of information to improve recommendation performance.

3 Methods

Our main objective is to leverage the full potential of deep representations of multimodal contextual information. The overall architecture is visually illustrated in Figure 2, and the exploration framework is demonstrated in Algorithm 1. This framework consists of two key modules: **Context-specific Encoders** (Section 3.2) illustrated in Figure 1, and **Contextual Bandit Policy** (Section 3.3).

*Authors in alphabetical order.

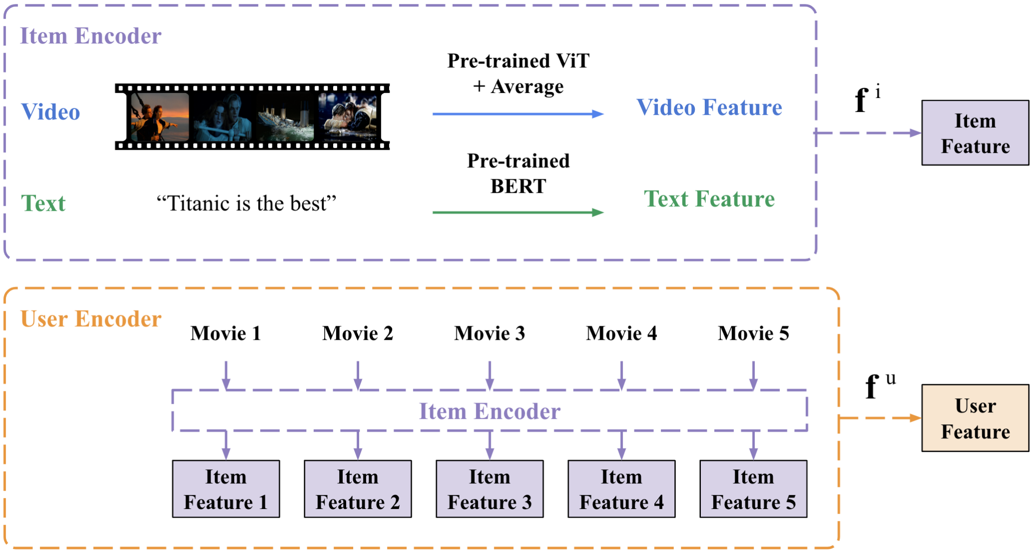


Figure 1: Context-specific Encoders To generate the contexts, we leverage the multi-modal features (e.g. video, text) and apply distinct fusion methods to both user and item embeddings.

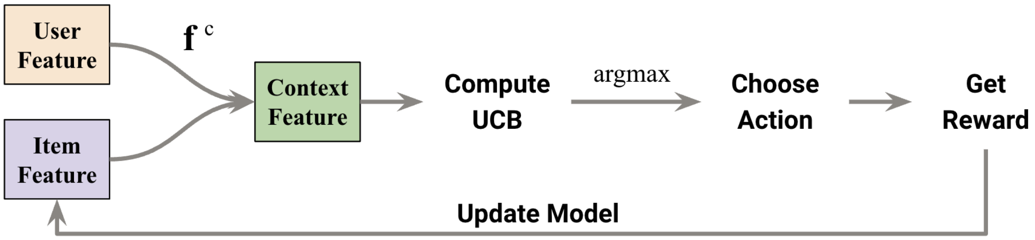


Figure 2: Multi-modal contextual bandits for movies recommendation We demonstrate the NeuralUCB algorithm[7] by leveraging context features derived from user and item embeddings.

3.1 Exploration Framework

Our framework is built upon the general bandit problem formulation. In this setup, the recommender system acts as an *agent*, selecting items (referred to as *arms*) based on user and item embeddings (the *context*). Using a policy, such as a greedy approach, the agent recommends items to the user. Feedback on user interaction is received, and the objective is to maximize cumulative rewards. The framework aims to achieve this goal by optimizing the policy over iterations. The specific procedures are outlined in Algorithm 1.

3.2 Context-specific Encoders

We apply distinct encoder modules to generate item and user embeddings, which collectively form the *context* in our framework. The item encoder leverages the visual and textual information of items, while the user encoder learns representations based on the items that users have previously watched. This approach allows us to effectively capture and integrate item-specific and user-specific details within the context.

Item Encoder consists of separate frozen modules for visual and textual representations. To encode the visual information, we employ a visual encoder that takes a video with a total of F frames as input. For each video, we randomly select F consecutive frames from the entire set. Then, we utilize a pre-trained Vision Transformer(**ViT**) [8], which is a transformer-based image encoder, to generate

Algorithm 1 NeuralUCB

```
Input: Number of rounds $T$, regularization parameter $\lambda$, exploration parameter $\nu$, confidence parameter $\delta$, norm parameter $S$, step size $\eta$, number of gradient descent steps $J$, network width $m$, network depth $L$.
Initialization: Randomly initialize $\boldsymbol{\theta}_0$ as described in the text
Initialize $\mathbf{Z}_0 = \lambda\mathbf{I}$
for $t = 1, \ldots, T$ do
    Observe $\{\mathbf{x}_{t,a}\}_{a=1}^{K}$
    for $a = 1, \ldots, K$ do
        Compute $U_{t,a} = f(\mathbf{x}_{t,a}; \boldsymbol{\theta}_{t-1}) + \gamma_{t-1}\sqrt{\mathbf{g}(\mathbf{x}_{t,a}; \boldsymbol{\theta}_{t-1})^\top \mathbf{Z}_{t-1}^{-1} \mathbf{g}(\mathbf{x}_{t,a}; \boldsymbol{\theta}_{t-1})/m}$
        Let $a_t = \mathrm{argmax}_{a \in [K]} U_{t,a}$
    end for
    Play $a_t$ and observe reward $r_{t,a_t}$
    Compute $\mathbf{Z}_t = \mathbf{Z}_{t-1} + \mathbf{g}(\mathbf{x}_{t,a_t}; \boldsymbol{\theta}_{t-1})\mathbf{g}(\mathbf{x}_{t,a_t}; \boldsymbol{\theta}_{t-1})^\top/m$
    Let $\boldsymbol{\theta}_t = \mathrm{TrainNN}(\lambda, \eta, J, m, \{\mathbf{x}_{i,a_i}\}_{i=1}^{t}, \{r_{i,a_i}\}_{i=1}^{t}, \boldsymbol{\theta}_0)$
    Compute
        $\gamma_t = \sqrt{1 + C_1 m^{-1/6}\sqrt{\log m}L^4 t^{7/6}\lambda^{-7/6}} \cdot \Big(\nu\sqrt{\log\frac{\det \mathbf{Z}_t}{\det \lambda\mathbf{I}} + C_2 m^{-1/6}\sqrt{\log m}L^4 t^{5/3}\lambda^{-1/6} - 2\log\delta} + \sqrt{\lambda}S\Big)$
            $+ (\lambda + C_3 tL)\Big[(1 - \eta m\lambda)^{J/2}\sqrt{t/\lambda} + m^{-1/6}\sqrt{\log m}L^{7/2}t^{5/3}\lambda^{-5/3}(1 + \sqrt{t/\lambda})\Big].$
end for
```

embeddings for each frame. We compute the average mean of the frame embeddings, resulting in the final visual representation for the video denoted as $i_{j_{\text{video}}} \in \mathbb{R}^d$.

For the textual encoder, we employ a pre-trained Bidirectional Encoder Representations from Transformers(**BERT**) [9], which is a transformer-based text encoder. To process the textual information, we append a special [CLS] token to a sequence of W words for each item. The extended sequence is used as the input for the textual encoder. The textual encoder contextualizes the word embeddings within the entire text, and we extract the output embedding corresponding to the [CLS] token as the final textual representation denoted as $i_{j_{\text{text}}} \in \mathbb{R}^d$.

In order to obtain the ultimate representation for each item, we experiment with two fusion functions. The first function involves a straightforward addition operation, where i_j is obtained by adding $i_{j_{\text{video}}}$ and $i_{j_{\text{text}}}$. The second function utilizes concatenation, where i_j is obtained by combining $i_{j_{\text{video}}}$ and $i_{j_{\text{text}}}$ using the concatenation operation.

User Encoder In this study, we investigate the generation of user representations based on a set of item embeddings, denoted as X_k. This set, representing the collection of item embeddings corresponding to the items observed by user u_t, is defined as $X_k = x_1, x_2, ..., x_k$.

We explored three distinct operations to derive the user representation. Firstly, we employ the averaging operation, where the item embeddings in X_k are averaged to obtain the representation. Secondly, we utilize the last item embedding in X_k to capture the user's most recent preference. Lastly, we examine the direct use of the sequence X_k without applying any fusion function or transformation.

Context Encoder generates final context features by concatenating the item and user embeddings made from item encoder (Sec. 3.2) and user encoder (Sec. 3.2)

It is worth noting that when utilizing the direct sequence of items X_t for user representation, an attention operation is employed to contextualize item embeddings with user information. In this operation, the item feature of the arm is used as the query, while the item features within X_t are used as the key and value.

PCA is employed to decrease the dimensionality of item and user vectors, considering the significant memory and computational resources required for computing the inverse matrix, as explained in section 3.3. Subsequently, we utilized a previously established approach[3] for dimensionality reduction by performing K-Means clustering on the item features derived from the reduced space obtained through PCA. This clustering process grouped the items into N clusters. To determine the membership of each user within these N clusters, we calculated the Euclidean distance and applied a softmax function, ensuring that the sum of distances for each user vector equates to 1. This resulted in the reduction of user vectors to N dimensions, with each dimension corresponding to the membership of the respective group. The selection of N, which represents the number of clusters, is a hyperparameter.

3.3 Multi-modal Contextual Bandit

Following [7], we adopt Algorithm 1, NeuralUCB, as our model for multi-modal contextual bandits. It utilizes neural networks to ensure generalization and efficient exploration. User and item features extracted from each encoder are used to approximate the reward function using MLPs. Finally, we perform parameter updates through the neural network. This approach is expected to be effective in training the model with contextual information. However, it also requires significant computational cost due to the computation of matrix inverses. Therefore, it is crucial to understand the characteristics of the model and find agile solutions to tackle these challenges.

4 Experiments

4.1 Datasets

We use a popular movie recommendation dataset for our experiments: MovieLens 25M [10]. We choose the movie domain because it provides rich contextual information, such as trailer videos, textual summaries, and metadata such as genre and actors. This allows us to conduct extensive experiments to verify our model architecture and assess the impact of context features on performance.

To enhance the MovieLens dataset, we have made significant progress in collecting additional content data. Given that the original dataset provides only the metadata, such as genre, we take the following steps to gather visual and textual information of the items.

For visual contents, we use movie trailers provided by MovieLens [11] and MovieNet[2], since the full videos are publicly unavailable for most movies due to copyright. From each video, frames of size 224×224 are sampled at 2 fps. We drop the first and last 10% of the sampled frames, since they often include age rating screen or ending credits. The average length of the trailers is 137 seconds, so we get around 220 frames per video.

For text contents, we use movie synopsis collected from IMDB[3] for MovieLens. These synopses are 2–3 sentences that summarize the movie overview. The sentences are first tokenized at word level with the maximum length of 512, using uncased $\text{BERT}_{\text{BASE}}$ tokenizer [9] with $|V| = 30,522$. The average number of tokens in text contents is 54.7.

4.2 Evaluation Method

To evaluate the performance of our proposed bandit algorithm π for arm selection, we use offline data collected previously using a different policy. Particularly, to mitigate selection bias and improve the simulator when learning from logged data, we employ a widely-used approach for off-policy evaluation of bandit algorithms, known as the Intermediate Bias Mitigation Step via the Inverse Propensity Score (IPS) simulator [12], following [2]. This approach involves re-weighting the training samples using the inverse propensity score. The IPS is learned from the logged data using logistic regression. To convert predicted scores($\hat{y}$) from the simulator into binary rewards $\{0, 1\}$, we select a threshold that maximizes the f-score on a validation dataset.

4.3 Metrics

We divide the dataset into train and test data for fair evaluation. To measure the performance, we employ a widely used metric, *click through rate* (CTR) on the test dataset. We calculate CTR for each user u_i as $\mathit{CTR}_{u_i} = \frac{1}{T}\sum_{\tau=1}^{T} \mathbb{1}\{r^{\tau}_{u_i} = 1\}$, where τ represents the number of trials. $r^{\tau}_{u_i}$ represents the simulator's reward of the recommended item for user u_i at trial τ. To assess the overall performance of bandits policies, we consider the cumulative CTR over all users in test dataset $\sum_{i=1}^{U_{test}} \mathit{CTR}_{u_i}$.

[2] https://movienet.github.io/

[3] https://www.imdb.com/

5 Results

Table 1: Comparison of context-specific encoders for different feature dimension reduction methods on MovieLens 25M. (CTR)

Item Encoder	PCA (components = 4)			Clustering (clusters = 8)		
	User Encoder-Avg	User Encoder-Last	User Encoder-Seq	User Encoder-Avg	User Encoder-Last	User Encoder-Seq
Genre	11.1%	11.36%	**10.45%**	10.65%	**11.4%**	10.45%
Sum	**11.23%**	**11.4%**	10%	**10.7%**	9.75%	10.45%
Concat	11.17%	11.17%	3.6%	10.35%	9.8%	9.75%
Image	10.1%	11.09%	9.7%	10.1%	9.95%	**10.55%**
Text	10.69%	11.32%	3.75%	9.4%	11.2%	9.9%

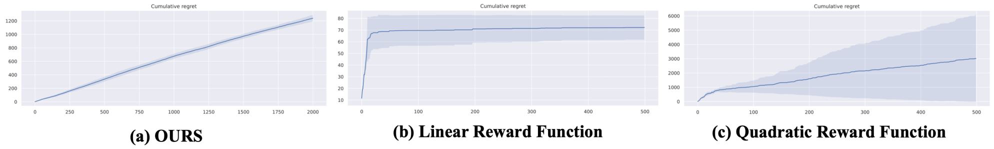
Table 1 reports the performance of our approach on the dataset MovieLens 25M. We analyze the impact of feature dimension reduction methods and context-specific encoders on performance. Additionally, we compare our multi-modal context creation method with unimodal approaches (i.e. image or text) as well as existing simple categorical (i.e. genre).

Regarding feature dimension reduction methods, the overall results demonstrate that the PCA method outperforms the clustering method. This suggests that the PCA method is more suitable for identifying the most important components and creating optimal combinations for the new embeddings in the recommendation task.

When comparing different user encoders with PCA method, the *User Encoder-Last* achieves the highest performance, suggesting that the most recent item watched by the user is a crucial factor in representing user preferences. For the clustering method, the highest performance varies depending on the item encoder, indicating the consistency and quality of features obtained through the PCA method. The *Item Encoder-Sum* generally performs well with PCA, while categorical data such as genre shows comparable performance. This highlights the effectiveness of human-labeled data for item features. However, the quality of multimodal features demonstrates that using only item data is sufficient to generate high-quality context features.

In terms of context creation, when using PCA method for better contexts, multi-modal context creation method (i.e. sum or concat) outperforms the use of unimodal information (i.e. image or text). This shows the importance of leveraging multiple modalities and considering a richer set of features for context creation in recommendation systems.

6 Discussion



(a) OURS **(b) Linear Reward Function** **(c) Quadratic Reward Function**

Figure 3: Regret analysis of different reward functions

Our methods demonstrate superior performance compared to the baseline method; however, we encountered instability during the algorithm training process. This suggests that our method did not operate optimally due to a lack of an ideal environment. Effective performance of the neural UCB algorithm relies on high-quality context features that accurately represent items and users, as well as a reward function learnable by the neural network. Stable convergence of total regret is achieved when using easily learnable linear or quadratic functions to represent the relationship between context features and rewards. In contrast, our method fails to converge, indicating potential issues with the quality of the context features or the reward function provided by the simulator, as depicted in Figure 3.

In conclusion, our research contributes to the exploration of multimodal inputs in contextual bandit problems. However, stable model training proved challenging. Future work could involve investigating advanced feature utilization, employing sophisticated neural networks instead of MLPs for better reward function approximation, and modifying the reward function using methods with less bias compared to the simulator.

References

[1] 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.

[2] Mengyan Zhang, Thanh Nguyen-Tang, Fangzhao Wu, Zhenyu He, Xing Xie, and Cheng Soon Ong. Two-stage neural contextual bandits for personalised news recommendation, 2022.

[3] 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*, 2010.

[4] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021.

[5] Hassan Akbari, Liangzhe Yuan, Rui Qian, Wei-Hong Chuang, Shih-Fu Chang, Yin Cui, and Boqing Gong. Vatt: Transformers for multimodal self-supervised learning from raw video, audio and text. *Advances in Neural Information Processing Systems*, 34:24206–24221, 2021.

[6] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks, 2019.

[7] Dongruo Zhou, Lihong Li, and Quanquan Gu. Neural contextual bandits with UCB-based exploration. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 11492–11502. PMLR, 13–18 Jul 2020.

[8] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2021.

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[10] 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.

[11] Sami Abu-El-Haija, Joonseok Lee, Max Harper, and Joseph Konstan. Movielens 20M youtube trailers dataset, 2018.

[12] Rubin D. Imbens, G. *Causal Inference for Statistics, Social, and Biomedical Sciences: An Introduction.* Cambridge University Press, 2015.