Solution to the Personalized Accommodation Review Ranking Task via Tabular Data Approach

Yu Tokutake[1]

[1]The University of Electro-Communications, 1-5-1 Chofugaoka, Chofu, Tokyo 182-8585, Japan

Abstract

This paper proposes a solution that won first place in the RecTour 2024 Challenge. The proposed solution employs a tabular data approach to address the review ranking task, comprising two stages: candidate generation, and ranking using the LightGBM model. The experimental results confirm that adjusting the number of negative samples in the training set, as an alternative to including all candidates in training, improves performance. In addition, the incorporation of text embeddings generated from user accommodation and review information as features further enhances accuracy. The code is available at https://github.com/ty1260/rectour2024_challenge.

Keywords

Review ranking, user review, personalization, tabular data, accommodation

1. Introduction

User-generated web reviews have become a widely used source of information for decision-making processes including product purchases, tourist attraction visits, and accommodation bookings. However, as the number of reviews increases, it becomes increasingly difficult for users to examine all available information. This has led to the emergence of personalized review ranking algorithms as a critical research field aimed at efficiently presenting useful reviews to users. Large public datasets for algorithm development include the Amazon Reviews dataset [1] in the e-commerce domain and the Yelp dataset[1] in the restaurant domain. In comparison to these datasets, the accommodation review dataset is on a smaller scale. The RecTour 2024 Challenge [2], a competition organized by Booking.com, aimed to address the task of personalized review ranking using a comprehensive dataset[2] [3] of accommodation reviews. Participants were tasked with developing algorithms that matched accommodation reviews with specific user and accommodation information. Prior to the commencement of the competition, Igebaria et al. [3] proposed a personalized accommodation review ranking approach that employs contrastive learning, evaluating it on the competition dataset.

This study presents a solution developed by the Ringo team, which won first place in the RecTour 2024 Challenge. The solution employs distinctive attributes of the supplied dataset through a tabular data approach, wherein the user, accommodation, and review data are combined to generate matching candidates. The challenging task is then formulated as a binary classification problem to determine matching candidates, and a LightGBM [4] classifier is deployed. In this process, features are extracted from users, accommodations, and reviews to be used as inputs for the classifier. Given the significant imbalance in the ground truth of all the generated candidates, the experiments evaluated changes in performance when the ratio of negative samples in the training set was manipulated. Additionally, as part of a natural language processing (NLP) approach, text embeddings of user accommodation and review information are generated and incorporated to enhance performance.

The remainder of this paper is organized as follows: Section 2 describes the dataset and evaluation metrics used in RecTour 2024 Challenge. Section 3 introduces the proposed method. Section 4 presents

Workshop on Recommenders in Tourism (RecTour 2024), October 18th, 2024, co-located with the 18th ACM Conference on Recommender Systems, Bari, Italy.

tokutakeyuu@uec.ac.jp (Y. Tokutake)
0009-0007-1399-1631 (Y. Tokutake)

[1] https://www.yelp.com/dataset

[2] https://huggingface.co/datasets/Booking-com/accommodation-reviews

Table 1
Statistics of RecTour 2024 Challenge dataset

	#users	#accommodations	#reviews
Training set	1,628,989	40,000	1,628,989
Validation set	203,787	5,000	203,787
Test set	199,138	5,000	199,138

the experimental results.

2. Challenge Task

2.1. Dataset

The competition dataset comprises three classes of data:

- Users: Information about users and accommodations.
- Reviews: Review texts for accommodations.
- Matches: Combinations of review texts (user_id, accommodation_id, and review_id) generated by users for accommodations.

User features include the type of stay (guest_type), guest's country of origin (guest_country), number of nights (room_nights), and month of stay (month) . Accommodation features include the type of accommodation (accommodation_type), country in which the accommodation is located (accommodation_country), average review rating of the accommodation (accommodation_score), rating by an external agency (accommodation_star_rating), and whether the accommodation is located on a beach (location_is_beach), on a ski resort (location_is_ski), or in a city (location_is_city_center). Review features include the title (review_title), positive section content (review_positive), and negative section content (review_negative) as textual data; the overall rating of the accommodation (review_score); and the number of users who referred to the review (review_helpful_votes).

The dataset was divided into training, validation, and test subsets. The statistics for each set are presented in Table 1. All users and reviews were distinct within the entire set, and there was always a one-to-one correspondence between users and reviews. Each accommodation had at least 10 reviews, and there were no common accommodations among the sets. The test set did not include match data, as the competition objective was to predict matching between users, accommodations, and reviews in the test set. Participants were required to rank reviews for each user accommodation using the prediction results and submit the top 10 reviews.

2.2. Evaluation Metrics

The top 10 predicted reviews were evaluated to determine whether they matched the reviews generated by users. To accomplish this, the mean reciprocal rank (MRR) and precision were used as evaluation metrics. MRR@k is calculated as follows:

$$\text{MRR@}k = \frac{1}{|U|}\sum_{u\in U}\frac{1}{\text{rank}_u}, \tag{1}$$

where U is the set of users and rank_u is the rank of the review generated by user u (if the review is not in the top-k, $1/(\text{rank}_u) = 0$). Precision@k is calculated as follows:

$$\text{Precision@}k = \frac{1}{|U|}\sum_{u\in U}\mathrm{I}\left[\text{rank}_u \leq k\right] \tag{2}$$

where $\mathrm{I}[\cdot]$ is the indicator function. In the field of information retrieval and recommendation, precision@k in Eq. (2) is used as a hit rate. For the competition task, k was set to 10.

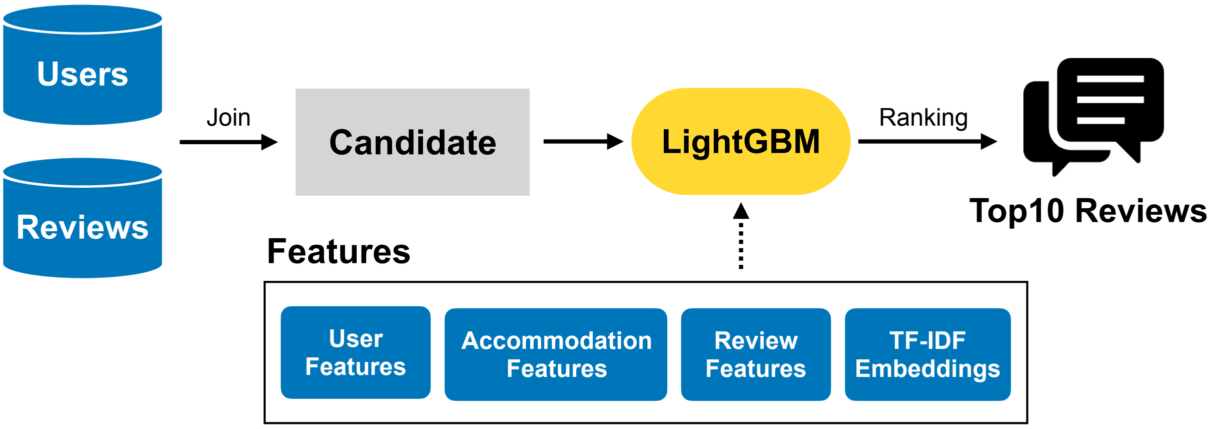

Figure 1: Overview of proposed approach

3. Method

3.1. Basic Strategy

An overview of the proposed approach is presented in Figure 1. The strategy employed for the competition task is a tabular data approach, wherein features are extracted from users, accommodations, and reviews to build a supervised model that predicts whether a review was generated by a user for a particular accommodation. First, candidate combinations of users, accommodations, and reviews are generated. Then, a binary classification model based on LightGBM is used to rank the reviews using the output probabilities as ranking scores. This approach is inspired by the two-stage recommendation approach that has been used in recent recommendation task competitions[3][4][5], which involves the generation and re-ranking of candidates for recommendation.

3.2. Candidate Generation

In a typical recommendation task, the number of users and items is so large that it is impractical to generate all user-item combinations. However, in this task, given information regarding accommodations stayed in by users and the reviews generated for each accommodation, it is feasible to generate all possible combinations of users, accommodations, and reviews. Specifically, candidates were generated by combining user and review data using accommodation_id as a key. Furthermore, the matched data were merged with these candidates for both the training and validation sets, and the existence of a user-generated review for a given accommodation was assigned as the ground truth. This information constitutes the fundamental input data for the binary classification model. The statistics of the generated candidates are listed in Table 2. As seen in the table, the candidates exhibited a pronounced imbalance, with a markedly greater number of negative samples than positive samples. The appropriate handling of the negative samples is therefore essential during model training. To address this issue, the negative samples in the training set were randomly downsampled. In the experiment described in Section 4, changes in performance were observed with respect to the ratio of negative samples.

3.3. Features for Review Ranking

The features used as inputs to the classification model were derived from information contained in the original data, as presented in Section 2.1. Furthermore, features based on accommodation and review were added as inputs. For accommodation-based features, aggregate features, such as the frequency of

[3] https://www.aicrowd.com/challenges/amazon-kdd-cup-23-multilingual-recommendation-challenge
[4] https://www.kaggle.com/competitions/otto-recommender-system
[5] https://www.kaggle.com/competitions/h-and-m-personalized-fashion-recommendations

Table 2
Statistics of generated candidates. Because the number of positives always matches the number of users and reviews, the numbers of positive and negative samples in the test set can be calculated.

	#candidates	#positive	#negative	positive:negative
Training set	214,311,737	1,628,989	212,682,748	1:131
Validation set	29,676,751	203,787	29,472,964	1:145
Test set	24,066,438	199,138	23,867,300	1:120

accommodation occurrences and average accommodation score, were added. The following review-based features were added:

- Review length: Number of words in review text (review_title, review_positive, review_negative).
- Sentiment analysis score for review title: A RoBERTa-based sentiment analysis model [5, 6] was employed to calculate sentiment scores (positive, negative, neutral) for the review titles. Each score ranges between 0 and 1.

Furthermore, as part of the NLP approach, term frequency-inverse document frequency (TF-IDF) was employed to generate text embeddings for user accommodation and review data. The texts to be embedded were concatenated in the format "<field_name>:<field_value>\n," following the method used by Igebaria et al [3]. The user and accommodation fields were arranged in the following order: guest_type, guest_country, room_nights, month, accommodation_type, accommodation_country, accommodation_star_rating, accommodation_score, location_is_ski, location_is_beach, location_is_city_center. The review fields were ordered as follows: review_title, review_positive, review_negative, review_score. The TF-IDF model was trained using the entire training, validation, and testing sets. Because the resulting embeddings were large, with 2,031,914 and 238,788 dimensions, ICA [7] was applied to reduce the dimensionality to 100. The resulting TF-IDF embeddings were used as features in one of the model variations.

4. Experiment

The effectiveness of the proposed method was evaluated using the test set. For comparison, RAND, which randomly selects 10 reviews for each user from all possible candidates (see Table 2), and Helpful Votes, which select the top 10 reviews from the candidates based on review_helpful_votes, were employed as baselines. First, the proposed method's performance was evaluated without using TF-IDF embeddings by varying the number of negative candidates in the training set. In particular, performance was evaluated using a positive-to-negative ratio of $1 : n$, where n was set to $\{1, 2, 10, 15, 20, 25, 30, 131\}$. In this instance, $n = 131$ represents a scenario in which all negative candidates were utilized without downsampling. Subsequently, the performance of the proposed method with $n = 20$, which showed the best performance, was evaluated using the TF-IDF embeddings as features. Table 3 presents the MRR@10 and Precision@10 results for the leaderboard across all methods, where the proposed method is denoted as the LGBM. As observed from the table, the proposed method outperformed both RAND and Helpful Votes in all variations. Additionally, by adjusting n, MRR@10 improved by up to 0.0078 points and precision@10 improved by up to 0.0188 points compared to the case of $n = 131$. Furthermore, the use of TF-IDF embeddings resulted in an improvement of 0.0665 points for MRR@10 and 0.1171 points for precision@10.

5. Conclusion

This study proposes a solution for the RecTour 2024 Challenge that employs a tabular data approach, comprising a two-stage process: candidate generation and ranking. By appropriately adjusting the

Table 3
Comparison of MRR@10 and precision@10 on leaderboard of test set

Method	MRR@10	Precision@10
RAND	0.0737	0.2515
Helpful Votes	0.0735	0.2511
LGBM w/o TF-IDF (n = 131)	0.0919	0.2892
LGBM w/o TF-IDF (n = 1)	0.0958	0.2997
LGBM w/o TF-IDF (n = 2)	0.0978	0.3035
LGBM w/o TF-IDF (n = 10)	0.0984	0.3047
LGBM w/o TF-IDF (n = 15)	0.0997	0.3079
LGBM w/o TF-IDF (n = 20)	0.0997	0.3080
LGBM w/o TF-IDF (n = 25)	0.0997	0.3079
LGBM w/o TF-IDF (n = 30)	0.0995	0.3071
LGBM (n = 20)	**0.1662**	**0.4251**

number of negative samples within the training set and incorporating text-embedding features, the proposed method improves prediction accuracy.

In future studies, two potential directions can be considered. The first direction entails improving the downsampling process in candidate generation for the proposed method. The current approach adopts random downsampling is applied, which may introduce bias to the training data for certain users and accommodations. Therefore, it is necessary to explore more appropriate downsampling techniques. The second direction involves comparison and integration with other NLP approaches, such as feature generation based on fine-tuning language models and ensemble methods for the prediction results.

Acknowledgments

I would like to thank Dr. Kazushi Okamoto for his help in providing computational resources and writing this paper.

References

[1] Y. Hou, J. Li, Z. He, A. Yan, X. Chen, J. McAuley, Bridging Language and Items for Retrieval and Recommendation, 2024. doi:10.48550/arXiv.2403.03952.

[2] A. Livne, E. Fainman, Booking.com RecTour 2024 Challenge, https://workshops.ds-ifs.tuwien.ac.at/rectour24/rectour-2024-challenge/, 2024. In ACM RecSys RecTour '24, October 18th, 2024, Bari, Italy.

[3] R. Igebaria, E. Fainman, S. Mizrachi, M. Beladev, F. Wang, Enhancing Travel Decision-Making: A Contrastive Learning Approach for Personalized Review Rankings in Accommodations, 2024. doi:10.48550/arXiv.2407.00787.

[4] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, T.-Y. Liu, LightGBM: A Highly Efficient Gradient Boosting Decision Tree, in: Advances in Neural Information Processing Systems, volume 30, Curran Associates, Inc., 2017. URL: https://proceedings.neurips.cc/paper_files/paper/2017/file/6449f44a102fde848669bdd9eb6b76fa-Paper.pdf.

[5] J. Camacho-collados, K. Rezaee, T. Riahi, A. Ushio, D. Loureiro, D. Antypas, J. Boisson, L. Espinosa Anke, F. Liu, E. Martínez Cámara, TweetNLP: Cutting-Edge Natural Language Processing for Social Media, in: Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Association for Computational Linguistics, 2022, pp. 38–49. doi:10.18653/v1/2022.emnlp-demos.5.

[6] D. Loureiro, F. Barbieri, L. Neves, L. Espinosa Anke, J. Camacho-collados, TimeLMs: Diachronic Language Models from Twitter, in: Proceedings of the 60th Annual Meeting of the Association for

Computational Linguistics: System Demonstrations, Association for Computational Linguistics, 2022, pp. 251–260. doi:10.18653/v1/2022.acl-demo.25.

[7] A. Hyvärinen, E. Oja, Independent Component Analysis: Algorithms and Applications, Neural Networks 13 (2000) 411–430. doi:10.1016/S0893-6080(00)00026-5.