# Condition-Aware Cultural Heritage Recommender System Using Knowledge Graph


Yeana Cha*
Department of Mathematics
Konkuk University
Seoul, South Korea
chayn02@konkuk.ac.kr

Yoojin Kim*
Department of Mathematics
Konkuk University
Seoul, South Korea
jina020131@konkuk.ac.kr

Jaeseung Kim*
Department of Mathematics
Konkuk University
Seoul, South Korea
dgd05037@konkuk.ac.kr

Jiwon Lee
Hyper-reality Metaverse Research Laboratory
Electronics and Telecommunication Research Institute
Daejeon, South Korea
ez1005@etri.re.kr

Hwijae Son
Department of Mathematics
Konkuk University
Seoul, South Korea
hwijaeson@konkuk.ac.kr



***Abstract*—This study proposes a novel approach to cultural heritage recommendation. Unlike conventional systems that focus only on individual entities, our method redefines the knowledge graph triple, consisting of head, relation, and tail, as a unified semantic unit to capture relational context and structural semantics. This enables a condition-aware similarity metric that yields more accurate, explainable, and personalized cultural heritage recommendations.**



***Keywords—Knowledge graph, Knowledge graph attention network, Condition-aware recommendation***


## I. Introduction

Knowledge graphs enable semantic modeling of cultural heritage data. However, existing knowledge graph-based recommender systems [1] rely on similarity between individual entities, thereby tend to easily overlook relational semantics.

To address this limitation, we propose a novel similarity measure that treats the entire triple as a unified semantic unit. The full structure of a triple is utilized in our method to enable more precise and condition-aware recommendations tailored to a user's specific interests. Moreover, it supports conditioning triples per head entity, allowing diverse relational conditions to be reflected in the recommendation process.

This flexibility allows the model to integrate diverse relational contexts, thereby enhancing the representation of user intent and enabling condition-aware, semantically enriched recommendation.

## II. Methodology

### A. Datasets

We constructed a knowledge graph using data on 12,240 cultural heritage items, each represented as a node. Items are connected to related entity nodes (e.g., period, pattern, material) via labeled edges such as *hasPeriod* and *hasPattern*, based on 11 descriptive attributes (see Table I).

### B. Knowledge Graph Embedding

We construct the input for the knowledge graph attention network (KGAT) model [2] using triples of the form (h, r, t), where h and t represent the head and tail entities, and r denotes the relation between them. Each entity is modeled as a node, and each relation is represented as an edge connecting the nodes, forming a directed multigraph structure. All entities are embedded into vectors of size 256.

TABLE I. Cultural Heritage Datasets

| Collection Number | Period | Category | Pattern | Material |
|---|---|---|---|---|
| Painting 775 | Goryeo | Housing | Flower, Plants | Soft Material |
| Painting 4881 | Goryeo | Dietary Life | Broken Branch | Celadon |
| Painting 3726 | Joseon | Culture and Arts | Flower, Bird | - |

We model the relationships between cultural heritage items and their attributes and aim to recommend relevant items to visitors based on their individual interests. To capture structured connections, we adapt TransR [3] framework for condition-aware recommendation.

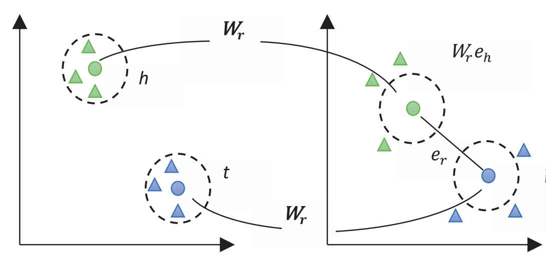



Fig. 1. Entity and relation embedding process in TransR.

As shown in Figure 1, TransR projects entities into relation-specific spaces using a projection matrix $W_r \in R^{d\times k}$. In this space, head and tail entities involved in a relation are drawn closer together, while unrelated entities become more distinguishable. This reflects the relational constraint in TransR, where the projected head entity and the relation vector are expected to approximate the projected tail entity, expressed as $W_r e_h + e_r \approx W_r e_t$.

This projection helps capture relation-specific roles of entities more effectively, which improves condition-aware

* These authors contributed equally to this work.

recommendation. Based on this structure, TransR defines the score function for a triple as:

$$g(h,r,t) = \left|\left|W_r e_h + e_r - W_r e_t\right|\right|_2^2 \tag{1}$$

The model is trained to minimize scores for positive triples $(h,r,t) \in G$, while maximizing scores for negative samples $(h,r,t) \notin G$. The model is optimized using the following loss function where $\sigma(x)$ denotes the sigmoid function:

$$L_{kg} = \sum_{(h,r,t)\in G,(h,r,t')\notin G} -\log\sigma\big(g(h,r,t) - g(h,r,t')\big) \tag{2}$$

*C. Condition-aware Similarity*

A condition-aware similarity measure is proposed to enable recommendation based on contextual triples. At inference, the model receives a user-specified condition triple, and computes similarity scores between this input and each candidate item $h$.

The condition-aware similarity function is derived from a modified interpretation of the TransR constraint, where the head entity is expected to approximate the combination of the relation vector and the tail entity. This constraint can be expressed as:

$$W_r e_h \approx W_r e_t - e_r \tag{3}$$

In this formulation, $W_r e_t - e_r$ is interpreted as a condition-aware representation. This is then compared with the transformed embedding of a candidate item $W_r e_h$ using condition-aware similarity:

$$similarity\ (h,r,t) = \frac{\langle W_r e_t - e_r, W_r e_h\rangle}{\|W_r e_t - e_r\|_2 \cdot \|W_r e_h\|_2} \tag{4}$$

To incorporate information from the head entity itself, we aggregate the head-only similarity with the condition-aware similarity through a weighted sum. The precise value of $\alpha \in [0,1]$ is determined empirically (see Section III).

$$similarity = \alpha \cdot similarity(h) + (1-\alpha) \cdot similarity(h,r,t) \tag{5}$$

When multiple triples $(h, r_i, t_i)$ are provided, the model computes the mean of their corresponding similarity scores:

$$mean\ similarity(h) = \frac{1}{N}\sum_{i=1}^{N} similarity(h, r_i, t_i) \tag{6}$$

In this case, Equation (5) is extended by aggregating the mean similarity in (6) with the head-only similarity term. This method allows the system to generate recommendations that are explicitly conditioned on relational inputs, rather than relying solely on general similarity between items.

## III. Experiments and Results

To evaluate the effectiveness of the proposed recommender system, we first compare recommendation performance when using only head entities and using only input triples. We then sweep $\alpha$ between 0 and 1 to identify the optimal trade-off.

Table II presents results using only the head entity *(Painting3070)* as input. In this setting, the model generates recommendations based on original cosine similarity. This suggests that the model primarily captures surface-level similarity, failing to reflect deeper attribute-level or contextual relationships.

TABLE II. Example of Recommendation Result Using H

| | Input | Recommendation | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| |  |  |  |  |
| Painting number | Painting 3070 | Painting 9877 | Painting 5174 | Painting 9230 |
| Pattern | - | Mountain, Tree, Theme | Mountain, Tree, Theme, Structure | Mountain, Structure |

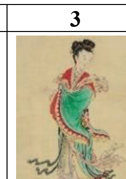
In contrast, Table III shows results based on the input triple *(Painting3070, Pattern, People).* The model generates recommendations that are better aligned with the condition. The results maintain Pattern relevance, while also demonstrating improved diversity.

Table IV presents results when using multiple conditions. In this setting, the model aggregates similarity scores across two condition triples: *(Painting3070, Pattern, People), (Painting3070, Pattern, Theme).* This enables the model to capture user intent more robustly and generate recommendations with richer attributes and relational contexts than those obtained from a single triple.

TABLE III. Example of Recommendation Result Using One Triple

| | Input | Recommendation | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| |  |  |  |  |
| Painting number | Painting 3070 | Painting 10505 | Painting 3703 | Painting 5409 |
| Pattern | People | People | People | People, Bird, Tree |

TABLE IV. Example of Recommendation Result Using Two Triples

| | Input | Recommendation | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| |  |  |  |  |
| Painting number | Painting 3070 | Painting 2651 | Painting 2291-13 | Painting 3822 |
| Pattern | People, Theme | People | People, Theme | People, Theme |

The previous experiments confirm the benefit of conditioning on relational triples. We now analyze the effect of the weighting parameter $\alpha$, which balances head-only similarity and condition-aware similarity, to identify the optimal setting.

TABLE V. IMPACT OF $\alpha$ ON RECOMMENDATION PERFORMANCE

| alpha | Precision@10 | NDCG@10 |
|---|---|---|
| 0.0 | 0.7665 | 0.9129 |
| 0.2 | 0.8047 | 0.9348 |
| 0.4 | 0.8284 | 0.9575 |
| 0.5 | 0.8282 | 0.9645 |
| 0.6 | 0.8209 | 0.9640 |
| 0.8 | 0.7879 | 0.9588 |
| 1.0 | 0.7081 | 0.9480 |

Table V summarizes user-free, ranking-based evaluation results. Precision@K measures the proportion of top-K ranked candidates that match the input condition (r, t) and NDCG@K evaluates ranking quality by assigning higher weights to condition-matching items placed at higher positions. As shown in Table V, we adopt $\alpha = 0.5$ as the final setting, which achieves the optimal balance between head-only similarity and condition-aware similarity. These metrics assess structural correctness with respect to the KG, not user relevance. The results indicate that either component alone is sub-optimal, whereas their weighted sum yields the strongest ranking performance.

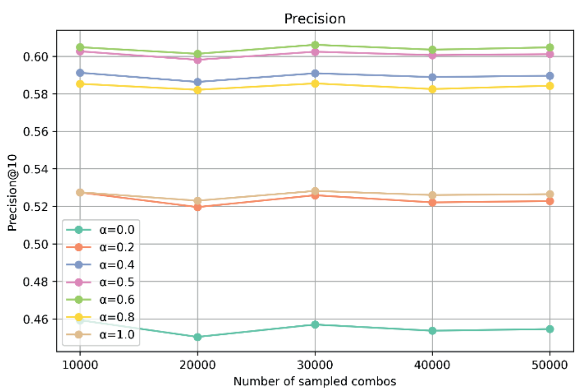



Fig. 2. Precision@10 across sampling sizes for different $\alpha$ values.

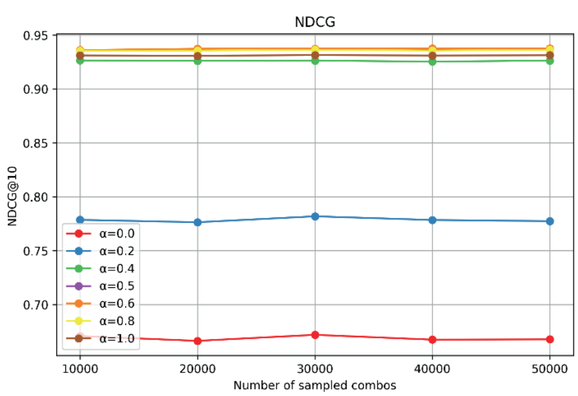



Fig. 3. NDCG@10 across sampling sizes for different $\alpha$ values.

For multiple input conditions, exhaustive experiments become computationally expensive due to the large number of possible combinations. Therefore, we perform sampling-based evaluations, gradually increasing the number of sampled combinations. As shown in Fig. 2 & Fig. 3, both Precision@K and NDCG@K stay stable over different $\alpha$ values, and the best performance is achieved when $\alpha = 0.6$.

Overall, these findings demonstrate the effectiveness of incorporating relational conditions into the recommendation process and validate the proposed framework.

## IV. CONCLUSION

In this study, we developed a cultural heritage recommender system that utilizes multiple triples as input, enabling the condition-aware recommendations. By effectively processing input triples, the system generates condition-aware recommendations aligned with users' specific interests and exploration goals regarding cultural heritage items. For future work, we aim to incorporate user data into the Collaborative Knowledge Graph (CKG) to enhance personalization in the domain of cultural heritage.

## ACKNOWLEDGMENT

This research was supported by Culture, Sports and Tourism R&D Program through the Korea Creative Content Agency(KOCCA) grant funded by the Ministry of Culture, Sports and Tourism(MCST) in 2023(Project Name: Development of storytelling AI technology for cultural heritage tailored to the various interests of users, Project Number: RS-2023-00220195, Contribution Rate: 100%)

## REFERENCES

[1] Jaeseung Kim, Jiwon Lee, and Hwijae Son. "Cultural Heritage Knowledge Graph and Recommender System." *Proceedings of the 2024 15th International Conference on Information Communication Technology Convergence (ICTC)*. IEEE, 2024, pp. 265–266.

[2] Wang, Xiang, et al. *"Kgat: Knowledge graph attention network for recommendation."* Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining. 2019.

[3] LIN, Yankai, et al. Learning entity and relation embeddings for knowledge graph completion. In: *Proceedings of the AAAI conference on artificial intelligence*. 2015.