Supplementary Material-SeeKRec: Toward Semantic-empowered Knowledge-aware Recommendation

No Author Given

No Institute Given

1 Theoretical Foundations of Semantic Distillation

In the context of semantic distillation, the goal is to compress the high-dimensional semantic features while preserving their most critical information. This can be achieved using Singular Value Decomposition (SVD), which effectively captures the principal components of the semantic embedding. The following theorem formalizes the optimality of low-rank approximations, specifically demonstrating that truncating the singular values provides the best approximation in both spectral and Frobenius norms, which directly supports the process of semantic distillation.

Theorem 1. *Let $A \in \mathbb{R}^{m\times n}$ have singular value decomposition*

$$A = U\,\mathrm{diag}(\sigma_1, \dots, \sigma_r, 0, \dots, 0)\,V^T,$$

with $\sigma_1 \geq \cdots \geq \sigma_r > 0$ and $\mathrm{rank}(A) = r$. For $k < r$, define

$$A_k = U\,\mathrm{diag}(\sigma_1, \dots, \sigma_k, 0, \dots, 0)\,V^T.$$

Then A_k is a best rank-k approximation of A in both spectral norm and Frobenius norm, with errors

$$\min_{\mathrm{rank}(B)\leq k} \|A - B\|_2 = \sigma_{k+1},$$

$$\min_{\mathrm{rank}(B)\leq k} \|A - B\|_F = \sqrt{\sum_{i=k+1}^{r} \sigma_i^2}.$$

Proof. Write the full SVD

$$A = U\Sigma V^T, \tag{1}$$

$$\Sigma = \mathrm{diag}(\sigma_1, \dots, \sigma_r, 0, \dots, 0), \tag{2}$$

and let

$$\Sigma_k = \mathrm{diag}(\sigma_1, \dots, \sigma_k, 0, \dots, 0), \quad A_k = U\,\Sigma_k\,V^T. \tag{3}$$

Clearly $\mathrm{rank}(A_k) = k$ and

$$\|A - A_k\|_2 = \|\Sigma - \Sigma_k\|_2 = \sigma_{k+1}, \tag{4}$$

$$\|A - A_k\|_F^2 = \|\Sigma - \Sigma_k\|_F^2 = \sum_{i=k+1}^{r} \sigma_i^2. \tag{5}$$

Optimality in the spectral norm. Let B be any matrix with $\mathrm{rank}(B) \leq k$, and write its SVD

$$B = \widetilde{U}\,\widetilde{\Sigma}\,\widetilde{V}^T, \tag{6}$$

where $\widetilde{\Sigma}$ has at most k nonzero diagonal entries. Since $\mathrm{rank}(B) \leq k$, we have

$$\dim\bigl(\ker(B)\bigr) \;\geq\; n - k, \tag{7}$$

and from the SVD of A,

$$\dim\bigl(\mathrm{span}\{v_{k+1}, \ldots, v_n\}\bigr) \;=\; n - k. \tag{8}$$

Hence there exists a unit vector

$$x \;\in\; \ker(B) \;\cap\; \mathrm{span}\{v_{k+1}, \ldots, v_n\}, \tag{9}$$

so that

$$B\,x = 0, \qquad x = \sum_{i=k+1}^{n} \alpha_i\, v_i, \quad \sum_{i=k+1}^{n} \alpha_i^2 = 1. \tag{10}$$

Then

$$A\,x = \sum_{i=k+1}^{n} \alpha_i\, \sigma_i\, u_i, \tag{11}$$

and hence

$$\|A\,x\| = \Bigl\| \sum_{i=k+1}^{n} \alpha_i\, \sigma_i\, u_i \Bigr\| \;\geq\; \sigma_{k+1} \sqrt{\sum_{i=k+1}^{n} \alpha_i^2} = \sigma_{k+1}. \tag{12}$$

Define

$$y \;=\; \frac{A\,x}{\|A\,x\|}, \tag{13}$$

so that $\|y\| = 1$. Then by the variational characterization of the spectral norm,

$$\begin{aligned} \|A - B\|_2 &= \max_{\|u\|=\|v\|=1} v^T (A - B)\, u \\ &\geq y^T (A - B)\, x \\ &= y^T A\, x \\ &= \|Ax\| \\ &\geq \sigma_{k+1}. \end{aligned} \tag{14}$$

Finally, since for $A_k = U \Sigma_k V^T$ we have

$$\|A - A_k\|_2 = \sigma_{k+1}, \tag{15}$$

it follows that A_k attains this lower bound and is therefore optimal.

Optimality in the Frobenius norm. For any B with $\text{rank}(B) \leq k$,

$$\|A - B\|_F^2 = \sum_{i=1}^{\min(m,n)} \sigma_i(A - B)^2 \geq \sum_{i=k+1}^{r} \sigma_i(A)^2, \tag{16}$$

by the interlacing inequalities for singular values. Equality holds for $B = A_k$, since

$$\|A - A_k\|_F^2 = \sum_{i=k+1}^{r} \sigma_i^2. \tag{17}$$

Thus A_k minimizes both $\|A - B\|_2$ and $\|A - B\|_F$ over all matrices of rank at most k, with the stated minimal errors.

2 Prompt

2.1 User Preference Extraction

```
Based on the list of {field} that the user has liked, infer their
↪ preferences and list no more than five of the top preferences.
↪ Please separate the interests with commas.no further explanation is
↪ needed.
```

2.2 User Preference Alignment

```
You are an expert in user preference analysis and text processing. Your
↪ task is to aggressively merge similar preferences into broader
↪ categories.

Instructions:
1. Merge intents that share similar themes, domains, or purposes.
2. Create high-level categories (maximum 3-4 words) that encompass
↪ related intents.
3. Only keep intents separate if they belong to completely different
↪ domains.
4. Always prioritize merging; maerge rather than keep separate when in
↪ doubt.
5. Do not give any Explanation
6. Output must be in strict JSON format.

Output format:
{
    "merged_intents": [
        {
            "representative_intent": "short category name",
            "merged_from": ["intent1", "intent2", "intent3"]
        }
```

```
    ],
    "unchanged_intents": ["unique intent1", "unique intent2"]
}
```

2.3 Semantic Subgraph Pruning

```
You are helping to build a knowledge subgraph for a recommendation
↪  system.
Current Focus: {center_name} (a user)
Goal: Select relationships that help understand this user's preferences
↪  and interests to provide better recommendations
Available Relationships: {formatted_triplets)}
Selection Criteria: Choose relationships that are most informative for
↪  understanding user preferences.
Guidance:
Actual user interactions provide concrete behavioral evidence of
↪  preferences, while interests show potential preferences. For a
↪  comprehensive understanding:
(1)You MUST include at least 1 relationship showing actual user
↪  interaction with items (not just interests)
(2)Balance between behavioral evidence (interactions) and preference
↪  indicators (interests)
(3)Avoid selecting only interest-based relationships - mix them with
↪  interaction-based ones
Prioritize relationships that show proven user behavior alongside their
↪  stated preferences.
Output Format: Return only the numbers (1-{len(formatted_triplets)})
↪  separated by commas.
Example: "1,5,8,12,15"
Your selection:
```

```
You are helping to build a knowledge subgraph for a recommendation
↪  system.
Current Focus: {center_name} (an item)
Goal: Select relationships that help understand this item's
↪  characteristics and features to improve recommendation accuracy
Available Relationships {formatted_triplets}
Selection Criteria: Choose relationships that are most informative for
↪  understanding item features and characteristics.
Guidance: Consider relationships that reveal item attributes,
↪  categories, similar items, and descriptive features.
Output Format: Return only the numbers (1-{len(formatted_triplets)})
↪  separated by commas.
Example: "1,5,8,12,15"
Your selection:
```

2.4 Semantic Subgraph Evaluating

```
Evaluate the quality of this knowledge subgraph for recommendation
↪  purposes.
Target: {center_name} (a user)
Current Subgraph: {subgraph}
Question: Can we understand {center_name}'s preferences well enough for
↪  recommendations?
Evaluation Criteria:
(1) User interest coverage (books, genres, topics they like)
(2) Behavioral patterns (interaction history, preferences)
(3) Preference diversity (different types of interests)
Response: Only output "SUFFICIENT" or "INSUFFICIENT" - no explanation
↪  needed.
Your assessment:
```

```
Evaluate the quality of this knowledge subgraph for recommendation
↪  purposes.
Target: {center_name} (an item)
Current Subgraph {subgraph}
Question: Do we have enough information about {center_name} for
↪  recommendations?
Evaluation Criteria:
(1) Item characteristics (genre, author, topic, features)
(2) Item relationships (similar items, categories)
(3) Descriptive attributes (what makes this item unique)
Response: Only output "SUFFICIENT" or "INSUFFICIENT" - no explanation
↪  needed.
Your assessment:
```

2.5 Semantic Refining

```
Assume you are an expert in {field} recommendations. You will be
↪  provided with a user's interaction history and interest
↪  information, along with some higher-order relationships. Your task
↪  is to analyze and understand this user's preferences in order to
↪  build a user profile. From this profile, identify the most relevant
↪  information that will help recommend movies the user is likely to
↪  enjoy. Your response should focus on selecting key aspects of the
↪  user's preferences that are valuable for effective recommendations,
↪  and it should be expressed in a coherent paragraph with no more
↪  than 150 words.
```

```
Assume you are an expert in {field} recommendation. You will receive an
↪ item's description in natural language, along with its higher-order
↪ relationships. Your task is to deeply analyze and model the item's
↪ core attributes, and then select and emphasize the features most
↪ useful for matching this item to potential users. Present your
↪ findings as a single, coherent paragraph of no more than 200 words.
```

3 Details for Baselines

1) Knowledge-free Recommenders:

- LightGCN [3] propagates information across the interaction graph without feature transformation or nonlinear activation.
- SGL [8] introduces a self-supervised task to improve node representations and robustness, addressing noisy interactions.
- SimGCL [11] generates contrastive views by adding noise to the embedding space, improving representation distribution and mitigating popularity bias.

2) ID-based Knowledge-aware Recommenders:

- CKE [12] integrates TransR-based graph embeddings with collaborative filtering to improve item relationship modeling.
- CFKG [1] integrates knowledge-base embeddings with collaborative filtering to improve accuracy and generate personalized explanations.
- KGAT [6] integrates knowledge graph and interaction graph, employing attention mechanisms to propagate neighborhood information.
- KGIN [7] captures user interaction intent and integrates long-range semantic relationships from the knowledge graph.
- KGCL [10] employs self-supervised learning and graph enhancement to integrate knowledge graph information and mitigate noise.
- MCCLK [13] introduces a multi-level contrastive learning mechanism across collaborative, semantic, and structural graph views.
- KGRec [9] utilizes an innovative self-supervised approach to identify and strengthen valuable triples in the knowledge graph.
- CIKGRec [4] extracts ID-based user-side knowledge using an LLM, while enhancing robustness through a GMAE-based reconstruction module and cross-domain contrastive learning.

3) Semantic-based Knowledge-aware Recommenders:

- CoLaKG [2] leverages LLMs to comprehend both local and global knowledge graph information, enhancing item and user representation learning by integrating semantic embeddings with ID embeddings.

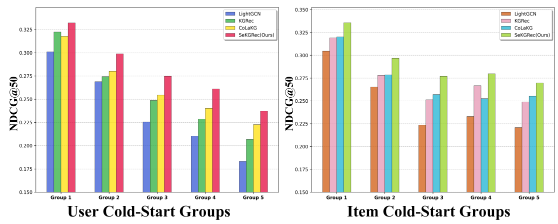

Fig. 1: Performance comparisions under different sparsity user groups and item groups in DBbook2014.

4 Cold-Start Scenario Analysis

One of the key motivation of introducing knowledge graphs is to alleviate cold-start problem [2]. To further explore the performance of SeeKRec in handling the cold-start problem, we conduct divide users and items into different groups of diffenent cold-start levels based on the training set following previous work [5]. We report the average NDCG@50 results for each group using our model and three representative baseline models. The results are shown in Figure 1, where moving rightward along the horizontal axis corresponds to increasing cold-start severity. The results demonstrate the superiority of SeeKRec across all groups, highlighting its advantage in addressing the cold-start problem. We attribute this superiority to our user semantic extraction module, which does not rely solely on limited interaction history but instead harnesses the reasoning capabilities and world knowledge of LLMs to achieve precise preference modeling. Furthermore, integrating LLM-enhanced semantic information as prior knowledge significantly improves recommendation performance in cold-start scenarios. Additionally, these findings further validate the effectiveness of the proposed semantic-aware MoE in fusing semantic information with collaborative signals.

References

1. Ai, Q., Azizi, V., Chen, X., Zhang, Y.: Learning heterogeneous knowledge base embeddings for explainable recommendation. Algorithms **11**(9), 137 (2018)
2. Cui, Z., Weng, Y., Tang, X., Lyu, F., Liu, D., He, X., Ma, C.: Comprehending knowledge graphs with large language models for recommender systems. In: Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 1229–1239 (2025)
3. He, X., Deng, K., Wang, X., Li, Y., Zhang, Y., Wang, M.: Lightgcn: Simplifying and powering graph convolution network for recommendation. In: Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval. pp. 639–648 (2020)
4. Hu, Z., Li, Z., Jiao, Z., Nakagawa, S., Deng, J., Cai, S., Zhou, T., Ren, F.: Bridging the user-side knowledge gap in knowledge-aware recommendations with large language models. In: Proceedings of the AAAI Conference on Artificial Intelligence. pp. 11799–11807 (2025)

5. Tang, G., Gan, X., Wang, J., Lu, B., Wu, L., Fu, L., Zhou, C.: Editkg: Editing knowledge graph for recommendation. In: Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 112–122 (2024)
6. Wang, X., He, X., Cao, Y., Liu, M., Chua, T.S.: Kgat: Knowledge graph attention network for recommendation. In: Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining. pp. 950–958 (2019)
7. Wang, X., Huang, T., Wang, D., Yuan, Y., Liu, Z., He, X., Chua, T.S.: Learning intents behind interactions with knowledge graph for recommendation. In: Proceedings of the web conference 2021. pp. 878–887 (2021)
8. Wu, J., Wang, X., Feng, F., He, X., Chen, L., Lian, J., Xie, X.: Self-supervised graph learning for recommendation. In: Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval. pp. 726–735 (2021)
9. Yang, Y., Huang, C., Xia, L., Huang, C.: Knowledge graph self-supervised rationalization for recommendation. In: Proceedings of the 29th ACM SIGKDD conference on knowledge discovery and data mining. pp. 3046–3056 (2023)
10. Yang, Y., Huang, C., Xia, L., Li, C.: Knowledge graph contrastive learning for recommendation. In: Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval. pp. 1434–1443 (2022)
11. Yu, J., Yin, H., Xia, X., Chen, T., Cui, L., Nguyen, Q.V.H.: Are graph augmentations necessary? simple graph contrastive learning for recommendation. In: Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval. pp. 1294–1303 (2022)
12. Zhang, F., Yuan, N.J., Lian, D., Xie, X., Ma, W.Y.: Collaborative knowledge base embedding for recommender systems. In: Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining. pp. 353–362 (2016)
13. Zou, D., Wei, W., Mao, X.L., Wang, Z., Qiu, M., Zhu, F., Cao, X.: Multi-level cross-view contrastive learning for knowledge-aware recommender system. In: Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval. pp. 1358–1368 (2022)