UNIFIED LOSS OPTIMIZATION FOR MULTI-RELATIONAL RECOMMENDATION

Alberto Ricchiuti
Department of Computer Science
University of Bari
Bari, Italy
a.ricchiuti20@studenti.uniba.it

August 26, 2025

Keywords CompGCN · Loss Optimization · ML-100k

ABSTRACT

Multi-relational graphs provide an expressive substrate for modern recommender systems, yet many architectures still disentangle knowledge-graph reasoning from user–item ranking into separate objectives. This work revisits the *Composition-based Graph Convolutional Network* (CompGCN) and the translational baseline TransE, integrating both into the `RecBole` ecosystem and, crucially, training them under a single loss that couples knowledge-graph embedding, Bayesian personalized ranking, and ℓ_2 regularization. Using MovieLens-100K enriched with side information, the paper presents a rigorous formulation of message passing, relation basis decomposition, edge normalization, and margin-based ranking. A detailed theoretical walk-through and an implementation blueprint—complete with annotated PyTorch code—enable transparent reproduction and fine-grained ablation of multi-relational recommendation models within a unified optimization landscape. The objective is to examine how such unification shapes representation learning, with empirical validation articulated in later sections.

1 Introduction

Graph Neural Networks (GNNs) have redefined the landscape of recommender systems by enabling end-to-end differentiable message passing among users, items, and auxiliary entities. Despite this progress, many GNN recommenders compress edge semantics into a single adjacency structure or optimize disjoint losses, typically one for link prediction in the knowledge graph and another for personalized ranking. Fragmented objectives can impede specialization, blur attributions, and complicate cross-model comparisons. This study investigates a principled unification of objectives on directed, typed graphs, asking how representation learning responds when embedding quality, recommendation accuracy, and weight decay are jointly optimized as a single scalar objective. The analysis considers two complementary models within an identical training loop and with shared negatives and parameters: a CompGCN variant that incorporates relation basis decomposition [8], configurable composition operators including circular correlation [5], and degree-normalized aggregation; and a minimalist TransE baseline implemented inside `RecBole` as a knowledge-graph embedding method for recommendation. The unified loss is introduced in Section 4.1, and results quantify how architectural choices translate into ranking improvements under shared optimization pressure.

2 Related Work

Translational representations such as TransE model each relation as a vector whose addition to a head entity approximates the tail in the embedding space [2]. The energy function $f_{\mathrm{KG}}(h, r, t) = \|\mathbf{E}_h + \mathbf{R}_r - \mathbf{E}_t\|_2$ is minimized for true triples and maximized for corrupted ones, providing a simple yet competitive baseline. Early GNN recommenders often treated user–item interactions as a homogeneous bipartite graph, later generalized to multi-relational settings through relation-specific transformations, attention, or composition mechanisms [4]. CompGCN composes node and relation embeddings to share parameters across relations while preserving expressiveness [9]. Knowledge-aware recommenders such as KGCN [10] and CKE [11] integrate KG-side information for improved ranking; CFKG [1] emphasizes explainability via heterogeneous embeddings; classical item-based heuristics (ItemKNN) remain competitive baselines [7]. Most prior works couple a recommendation objective such as BPR [6] with auxiliary terms using weighted sums or multi-stage training; comparatively fewer explore the effect of fully shared parameters and a single loss across both knowledge-graph and recommendation tasks.

3 Materials

The experiments use MovieLens-100K [3], which contains $|\mathcal{U}| = 943$ users, $|\mathcal{I}| = 1{,}682$ items, and 100,000 explicit ratings. The dataset is augmented with side information such as genres, directors, and actors, which are aligned to entities in an auxiliary knowledge graph. The resulting graph comprises an entity set $\mathcal{V}$ and a multi-relational edge set $\mathcal{E}$ consisting of typed, directed edges, together with inverse edges for each factual relation and self-loops to allow identity messages. The alignment procedure relies on normalized string matching of movie metadata, deterministic mapping to graph entity identifiers, programmatic construction of inverse relations, and the restriction to edges whose endpoints lie within the interaction universe to ensure consistency at training time. The `RecBole` framework [12, 13] standardizes dataset parsing, field mapping, negative sampling, training, and evaluation; by subclassing `KnowledgeRecommender`, a model can compute in a single step both a recommendation term over (u, i^+, i^-) and a margin-based knowledge-graph term over (h, r, t) and its negatives, with all parameters shared to produce a unified optimization landscape.

4 Methods

Consider a directed multi-relational graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with $|\mathcal{V}|$ entities and $|\mathcal{E}|$ labeled edges, where each edge is a quintuple $(s, d, r, \mathrm{dir}, \gamma)$ indicating source s, destination d, relation type r, direction $\mathrm{dir} \in \{\mathrm{out}, \mathrm{in}, \mathrm{loop}\}$, and a normalization scalar γ. Entity and relation embeddings of dimension D are arranged as $\mathbf{E} \in \mathbb{R}^{|\mathcal{V}| \times D}$ and $\mathbf{R} \in \mathbb{R}^{|\mathcal{R}| \times D}$.

A single CompGCN layer maps $(\mathbf{E}, \mathbf{R})$ to $(\mathbf{E}', \mathbf{R}')$ through composition, directional transformation, and normalized aggregation. Given an entity vector $\mathbf{e}_s$ and a relation vector $\mathbf{r}_r$, the composed message is

$$\phi(\mathbf{e}_s, \mathbf{r}_r) = \begin{cases} \mathbf{e}_s - \mathbf{r}_r & \texttt{sub} \\ \mathbf{e}_s \odot \mathbf{r}_r & \texttt{mult} \\ \mathbf{e}_s + \mathbf{r}_r & \texttt{add} \\ \mathrm{IFFT}\big(\mathrm{FFT}(\mathbf{e}_s) \odot \overline{\mathrm{FFT}(\mathbf{r}_r)}\big)/\sqrt{D} & \texttt{ccorr}, \end{cases} \tag{1}$$

where $\odot$ denotes the Hadamard product and the factor $1/\sqrt{D}$ follows the chosen FFT/IFFT normalization [5]. Directionality is modeled by a linear map $W_{\mathrm{dir}} \in \mathbb{R}^{D \times D}$ per direction, producing $\mathbf{m}_{s \to d, r} = \gamma\, W_{\mathrm{dir}}\, \phi(\mathbf{e}_s, \mathbf{r}_r)$ with symmetric normalization $\gamma = \deg_{\mathrm{in}}(d)^{-1/2} \deg_{\mathrm{out}}(s)^{-1/2}$. Node updates sum incoming messages and apply batch normalization, dropout, and a pointwise nonlinearity σ (ReLU in practice), yielding

$$\mathbf{e}'_d = \sigma\Big(\mathrm{Drop}\big(\mathrm{BN}\big(\textstyle\sum_{(s,r)} \mathbf{m}_{s \to d, r}\big)\big)\Big). \tag{2}$$

Relation embeddings are updated as $\mathbf{R}' = W_{\mathrm{rel}} \mathbf{R}$, and parameter efficiency is achieved via a basis decomposition $\mathbf{R} = \mathbf{C}\mathbf{B}$ with $\mathbf{C} \in \mathbb{R}^{|\mathcal{R}| \times b}$ and $\mathbf{B} \in \mathbb{R}^{b \times D}$, where $b \ll |\mathcal{R}|$ controls memory [8]. Stacking L layers yields $\mathbf{E}^{(L)}$ and $\mathbf{R}^{(L)}$ used for both knowledge-graph and recommendation scoring.

Listing 1: `CompGCNLayer` (excerpt)

```
def forward(self, ent_emb, rel_emb, edge_index, edge_type, edge_dir, edge_norm):
    src, dst = edge_index
    msg = self.compose(ent_emb[src], rel_emb[edge_type])
    out = torch.zeros_like(ent_emb)

    for dir_id, dir_name in enumerate(['out', 'in', 'loop']):
        mask = (edge_dir == dir_id)
        if mask.sum().item() == 0:
            continue
        W = self.W_dir[dir_name]
        norm = edge_norm[mask].unsqueeze(1)
        msg_dir = W(msg[mask]) * norm
        out = out.index_add(0, dst[mask], msg_dir)

    out = self.bn(out)
    out = self.act(self.dropout(out))
    rel_out = self.act(self.W_rel(rel_emb))
    return out, rel_out
```

Users and entities are embedded in a common space for the TransE baseline, using disjoint identifier ranges to avoid collisions. The knowledge-graph score adopts the original energy

$$s_{\mathrm{KG}}(h,r,t) = -\big\|\mathbf{E}_h + \mathbf{R}_r - \mathbf{E}_t\big\|_2, \tag{3}$$

reproduced in Listing 2.

Listing 2: TransE knowledge-graph score

```
def kg_score(self, h, r, t):
    h_e = self.entity_embedding(h)
    r_e = self.relation_embedding(r)
    t_e = self.entity_embedding(t)
    return -(h_e + r_e - t_e).norm(p=2, dim=1)
```

4.1 Unified Loss

The unified loss couples a margin-based knowledge-graph term with a BPR ranking term and an ℓ_2 penalty. Let $\mathcal{B}_{\mathrm{KG}}$ denote positive triples with negative tails generated by uniform tail corruption, and let $\mathcal{B}_{\mathrm{Rec}}$ denote user–item tuples for BPR. The objective

$$\begin{aligned}\mathcal{L} = & \frac{1}{|\mathcal{B}_{\mathrm{KG}}|}\sum_{(h,r,t)\in\mathcal{B}_{\mathrm{KG}}} \max\!\big(0,\ f(h,r,t) - f(h,r,t^-) + \gamma\big) \\ & + \ \alpha\frac{1}{|\mathcal{B}_{\mathrm{Rec}}|}\sum_{(u,i^+,i^-)\in\mathcal{B}_{\mathrm{Rec}}} -\ln\sigma\big(s_{ui^+} - s_{ui^-}\big) \ + \ \lambda\|\Theta\|_2^2 \end{aligned} \tag{4}$$

uses $f(h,r,t) = \|\mathbf{E}_h + \mathbf{R}_r - \mathbf{E}_t\|_2$ for TransE and $f(h,r,t) = \|\phi(\mathbf{E}_h, \mathbf{R}_r) - \mathbf{E}_t\|_2$ for CompGCN, together with the dot-product score $s_{ui} = \mathbf{u}_u^\top \mathbf{e}_i$. Unless otherwise stated, the mixing coefficient α remains fixed across epochs. Training proceeds in mini-batches by jointly sampling KG triples and recommendation tuples, computing all scores in a single forward pass, and backpropagating a single scalar objective.

5 Experimental Setup

The evaluation comprises BPR [6], CFKG [1], CKE [11], ItemKNN [7], KGCN [10], TransE [2], and CompGCN [9] under harmonized settings. Unless specified otherwise, optimization uses Adam with a learning rate of 10^{-3}; batch size is 512 for KG-driven models and 2048 for collaborative baselines; early stopping employs patience 10. KG models are trained for 30 epochs and baselines for 300 epochs. The embedding dimensionality is 128 for TransE and CompGCN and 64 for other models. Where applicable, the margin is 1.0 and the ℓ_2 regularization is 10^{-4}; CompGCN uses two layers, eight relation bases, a fixed mixing coefficient $\alpha = 0.5$, and gradient clipping with $\|g\|_2 \leq 1.0$. Ratings are binarized with a threshold of at least four stars. For each user, the most recent interaction forms the test item, the second most recent forms the validation item, and the remainder constitute the training set. Metrics are computed as full-ranking scores over the entire candidate set without sampled negatives and include Recall@10, NDCG@10, MRR@10, Hit@10, and Precision@10. All reported values are averaged over three random seeds with the same seeds reused across models; means and standard deviations are reported where appropriate. To summarize performance across metrics, the weighted score S is defined as the unweighted average of the five metrics, each already bounded in $[0, 1]$, and the best-epoch value refers to the epoch that maximizes S on the validation set. Experiments run on a single NVIDIA GPU with 12–16 GB of memory. Exact configuration files with library versions, random seeds, and preprocessing flags are provided in the accompanying repository, and the RecBole configuration specifying field mappings and evaluation settings is released for reproducibility.

Table 1: Core training hyperparameters by model (capacity & optimization).

Model	Emb Dim	LR	Batch	Epochs	Optim.	Layers
BPR	64	0.001	2048	300	Adam	—
CFKG	64	0.001	2048	300	Adam	—
CKE	64	0.001	2048	300	Adam	—
CompGCN	128	0.001	512	30	Adam	2
ItemKNN	—	0.001	2048	300	Adam	—
KGCN	64	0.001	2048	300	Adam	—
TransE	128	0.001	512	30	Adam	—

Table 2: Regularization and strategy hyperparameters by model.

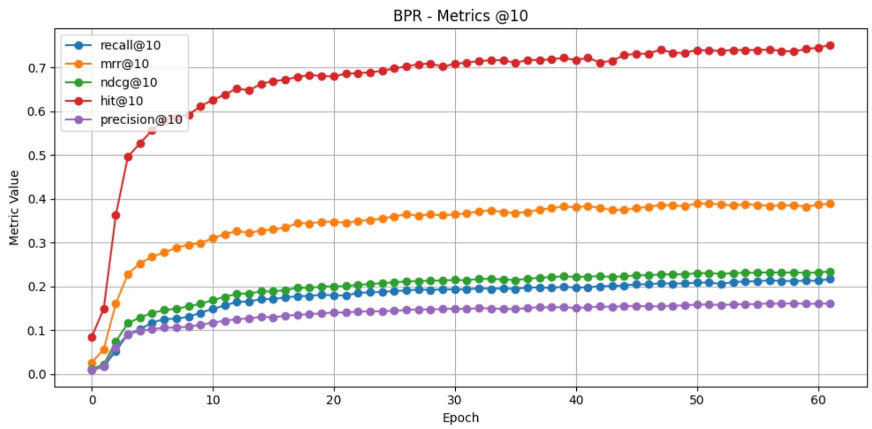
Model	Margin	L2 / Reg	Early Stop	α	Other
BPR	—	—	10	—	—
CFKG	1.0	—	10	—	—
CKE	—	[0.01, 0.01]	10	—	—
CompGCN	1.0	10^{-4}	10	0.5	`num_bases=8`; `clip_grad_norm=1.0`
ItemKNN	—	—	10	—	`shrink=0.0`; `knn_method=item`
KGCN	—	10^{-7}	10	—	`iter=1`; `aggr=sum`; `sample_size=4`
TransE	1.0	10^{-4}	10	0.5	`clip_grad_norm=1.0`

6 Results

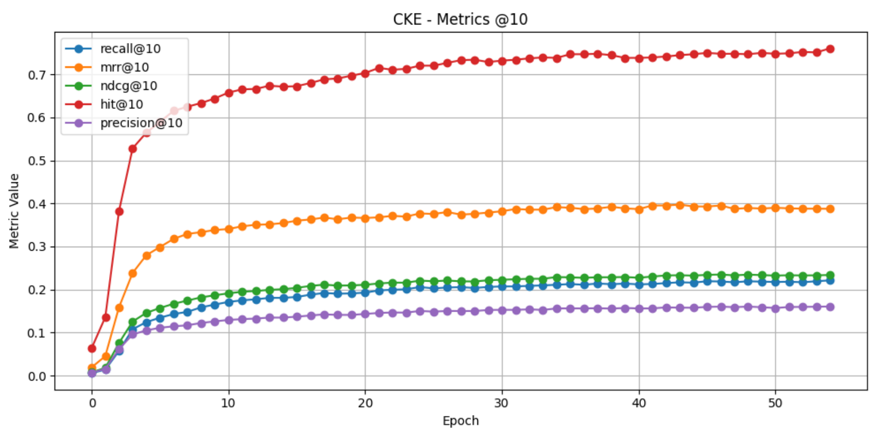
Table 3 summarize performance under the harmonized full-ranking protocol on ML-100K with temporal splits and three random seeds. Purely collaborative baselines (BPR, ItemKNN) provide a reasonable point of reference, with average scores in the 0.24–0.25 range. Classical knowledge-aware methods (CKE, KGCN) yield modest gains relative to these baselines but remain close in absolute terms. Within the unified-training setting considered here, *TransE* attains the highest mean score in our study (0.419 at the best validation epoch), accompanied by leading values on Recall@10 (0.437) and NDCG@10 (0.419). *CompGCN* variants are competitive overall, with the circular-correlation operator (`ccorr`) consistently surpassing the multiplicative and subtractive compositions (mean score 0.402; Recall@10 0.407; NDCG@10 0.399 at peak).[1]

Training curves indicate stable optimization dynamics for both *TransE* and *CompGCN*, with relatively rapid convergence and consistent plateaus across @10 metrics (Figure 3). Importantly, the observed gaps should be interpreted within the confines of the present capacity and optimization choices. Differences across *CompGCN* operators appear sensitive not only to the composition rule, but also to interactions among depth, number of relation bases, normalization, and regularization. Consequently, part of the empirical margin between models could plausibly reflect capacity allocation (e.g., embedding dimensionality, depth, and gradient clipping thresholds) rather than architectural principles alone. Overall, in this experimental regime *TransE* yields the strongest mean ranking performance, while *CompGCN* remains competitive and suggests room for improvement through targeted ablations and hyperparameter tuning.

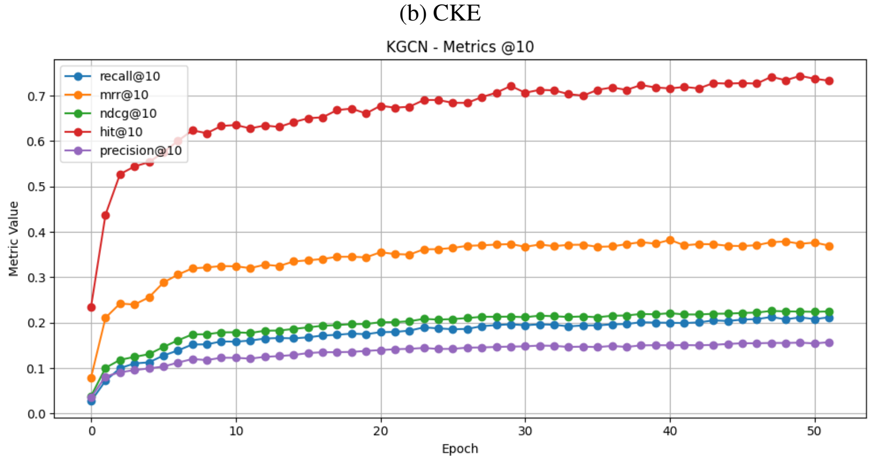
[1]For completeness: *CompGCN* (`mult`) and (`sub`) trail `ccorr` by several points on the aggregated score; CKE and KGCN track close to collaborative baselines. See Table 3.



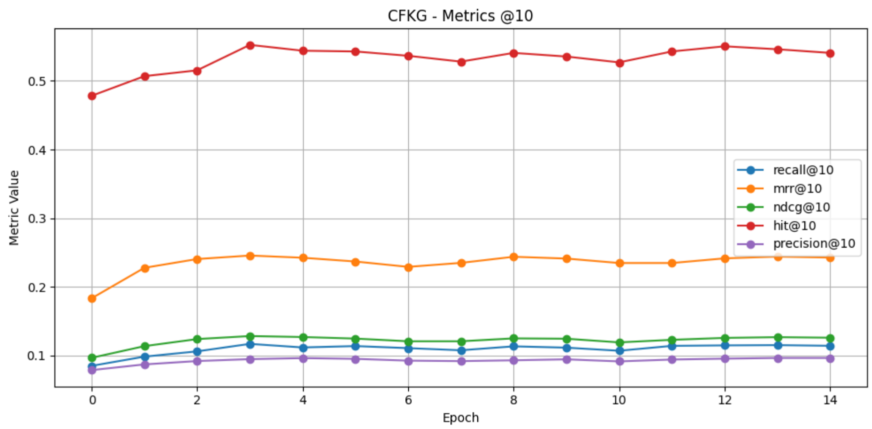
(a) BPR



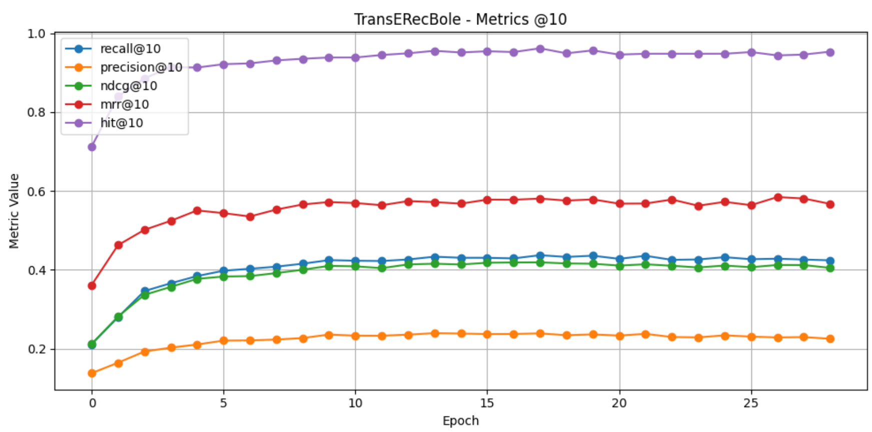
(b) CKE



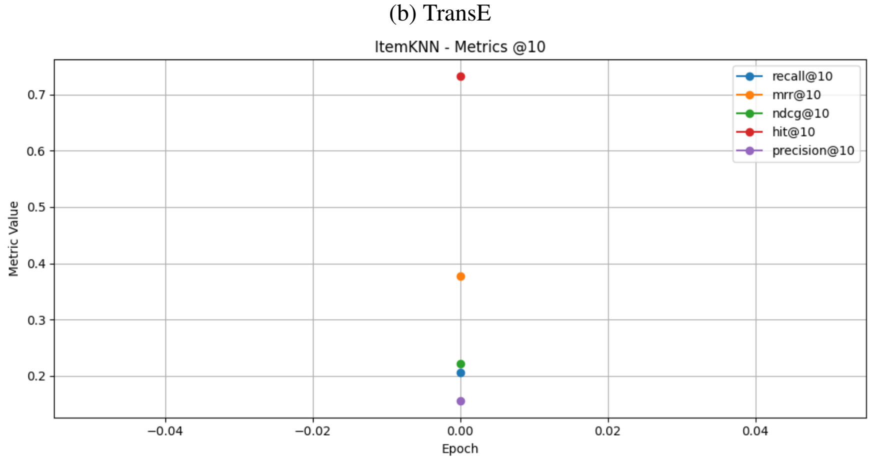
(c) KGCN



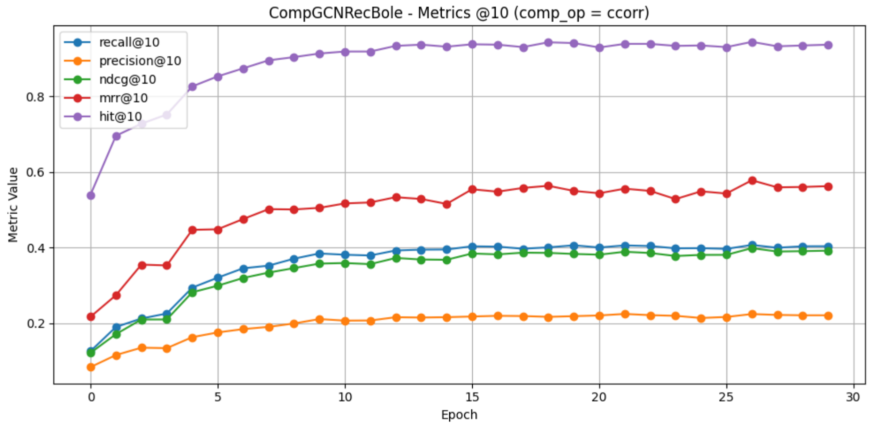
(a) CFKG



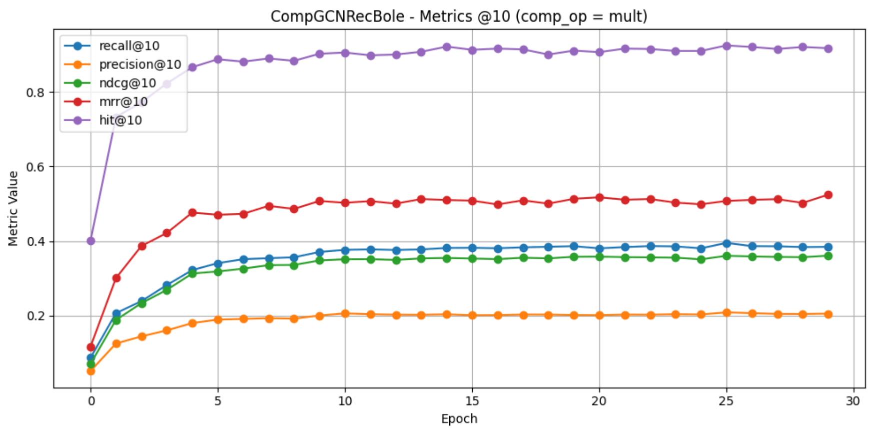
(b) TransE



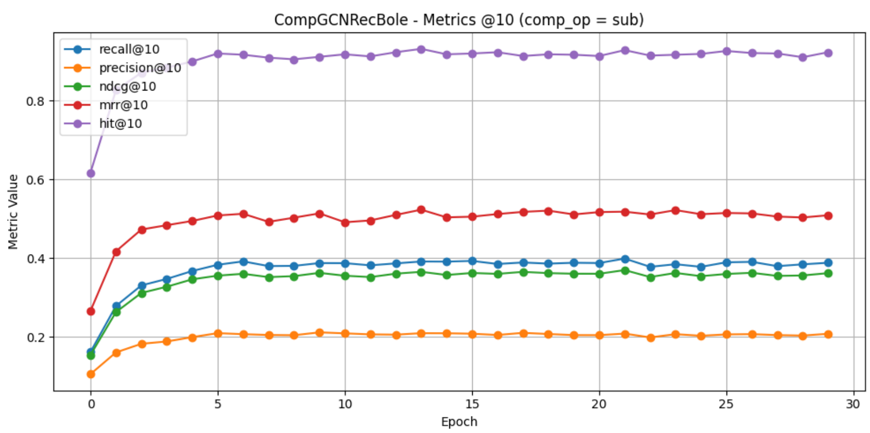
(c) ItemKNN



(a) CompGCN (ccorr)



(b) CompGCN (mult)



(c) CompGCN (sub)

Figure 3: Training curves (@10) for individual models.

Model	Best epoch	Avg. score	Recall	Precision	NDCG	MRR
BPR	61	0.25005	0.2168	0.1609	0.2340	0.3885
CFKG	3	0.146375	0.1169	0.0947	0.1283	0.2456
CKE	46	0.252275	0.2187	0.1603	0.2349	0.3952
CompGCNRecBole (ccorr)	26	0.402175	0.4072	0.2246	0.3988	0.5781
CompGCNRecBole (mult)	29	0.369100	0.3848	0.2058	0.3611	0.5247
CompGCNRecBole (sub)	21	0.373900	0.3991	0.2087	0.3698	0.5180
ItemKNN	0	0.240100	0.2058	0.1554	0.2213	0.3779
KGCN	47	0.242875	0.2126	0.1556	0.2260	0.3773
TransERecBole	17	0.419125	0.4373	0.2391	0.4192	0.5809

Table 3: Performance at the best epoch for each model.

7 Conclusions

This work presented an integrated implementation of *CompGCN* and *TransE* within `RecBole`, trained under a single objective that combines a margin-based knowledge-graph term, a BPR ranking component, and ℓ_2 regularization. On ML-100K, the empirical picture is nuanced: within the boundaries of our protocol and capacity choices, *TransE* tends to achieve the highest @10 metrics, whereas *CompGCN* delivers competitive results—particularly with `ccorr` composition—while retaining the modeling flexibility characteristic of relation-aware message passing. These findings are indicative rather than definitive; they do not warrant broad claims about structural superiority across datasets or scales.

Several limitations qualify the present conclusions. First, models differ in effective capacity (embedding size, depth), and we fixed the mixing coefficient α, precluding adaptive schedules. Second, we did not conduct extensive ablations on the number of relation bases, edge normalization strategies, or deeper GNN stacks. Third, the study is confined to a single domain and graph scale. Future work should therefore (i) explore dynamic or curriculum-based control of α, (ii) investigate hybrid composition mechanisms and attention within multi-relational layers, and (iii) assess scalability on larger, heterogeneous knowledge graphs with richer relation vocabularies. Taken together, the results delineate a promising yet cautious outlook: unified training is compatible with stable optimization and competitive top-k ranking in this setting, but a fuller account of the trade-offs between translational and convolutional multi-relational modeling will require broader ablations and cross-domain evaluations.

References

[1] Qingyao Ai, Vahid Azizi, Xu Chen, and Yongfeng Zhang. Learning heterogeneous knowledge base embeddings for explainable recommendation. *Algorithms*, 11(9), 2018. doi: 10.3390/a11090137. URL `https://www.mdpi.com/1999-4893/11/9/137`.

[2] Antoine Bordes, Nicolas Usunier, Alberto García-Durán, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. In *Advances in Neural Information Processing Systems 26 (NeurIPS 2013)*, pages 2787–2795. Curran Associates, Inc., 2013. URL `https://proceedings.neurips.cc/paper_files/paper/2013/file/1cecc7a77928ca8133fa24680a88d2f9-Paper.pdf`.

[3] F. Maxwell Harper and Joseph A. Konstan. The movielens datasets: History and context. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 5(4), 2015. doi: 10.1145/2827872. URL `https://dl.acm.org/doi/10.1145/2827872`.

[4] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR)*, 2017. URL `https://openreview.net/forum?id=SJU4ayYgl`.

[5] Maximilian Nickel, Lorenzo Rosasco, and Tomaso Poggio. Holographic embeddings of knowledge graphs. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence (AAAI-16)*, pages 1955–1961. AAAI Press, 2016. doi: 10.1609/aaai.v30i1.10314. URL `https://ojs.aaai.org/index.php/AAAI/article/view/10314`.

[6] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. Bpr: Bayesian personalized ranking from implicit feedback. In *Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 452–461. AUAI Press, 2009.

[7] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th International World Wide Web Conference (WWW)*, pages 285–295, Hong Kong, China, 2001. ACM. doi: 10.1145/371920.372071.

[8] Michael Schlichtkrull, Thomas N. Kipf, Peter Bloem, Rianne van den Berg, Ivan Titov, and Max Welling. Modeling relational data with graph convolutional networks. In *The Semantic Web – ESWC 2018*, volume 10843 of *Lecture Notes in Computer Science*, pages 593–607, Cham, 2018. Springer. doi: 10.1007/978-3-319-93417-4_38.

[9] Shikhar Vashishth, Soumya Sanyal, Vikram Nitin, and Partha Talukdar. Composition-based multi-relational graph convolutional networks. In *International Conference on Learning Representations (ICLR)*, 2020. URL `https://openreview.net/forum?id=BylA_C4tPr`.

[10] Hongwei Wang, Miao Zhao, Xing Xie, Wenjie Li, and Minyi Guo. Knowledge graph convolutional networks for recommender systems. In *Proceedings of the Web Conference (WWW)*, pages 3307–3313, 2019. doi: 10.1145/3308558.3313417.

[11] Fuzheng Zhang, Nicholas Jing Yuan, Defu Lian, Xing Xie, and Wei-Ying Ma. Collaborative knowledge base embedding for recommender systems. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 353–362, 2016. doi: 10.1145/2939672.2939673.

[12] Wayne Xin Zhao, Shanlei Mu, Yupeng Hou, Zihan Lin, Yushuo Chen, Xingyu Pan, Kaiyuan Li, Yujie Lu, Hui Wang, Changxin Tian, Yingqian Min, Zhichao Feng, Xinyan Fan, Xu Chen, Pengfei Wang, Wendi Ji, Yaliang Li, Xiaoling Wang, and Ji-Rong Wen. Recbole: Towards a unified, comprehensive and efficient framework for recommendation algorithms. In *Proceedings of the 30th ACM International Conference on Information and Knowledge Management (CIKM)*, pages 4653–4664, 2021. doi: 10.1145/3459637.3482016.

[13] Wayne Xin Zhao, Yupeng Hou, Xingyu Pan, Chen Yang, Zeyu Zhang, Zihan Lin, Jingsen Zhang, Shuqing Bian, Jiakai Tang, Wenqi Sun, Yushuo Chen, Lanling Xu, Gaowei Zhang, Zhen Tian, Changxin Tian, Shanlei Mu, Xinyan Fan, Xu Chen, and Ji-Rong Wen. Recbole 2.0: Towards a more up-to-date recommendation library. In *Proceedings of the 31st ACM International Conference on Information and Knowledge Management (CIKM)*, pages 4722–4726, 2022. doi: 10.1145/3511808.3557680.