Fairness and Bias in Recommender Systems

Alleviating the unfairness issue with knowledge-aware recommendation models

Yoan Popov[1]

Supervisor: Masoud Mansoury[1]

[1]**EEMCS, Delft University of Technology, The Netherlands**

A Thesis Submitted to EEMCS Faculty Delft University of Technology,
In Partial Fulfilment of the Requirements
For the Bachelor of Computer Science and Engineering
June 22, 2025

Name of the student: Yoan Popov
Final project course: CSE3000 Research Project
Thesis committee: Masoud Mansoury, Nergis Tömen

Abstract

This study investigates fairness in knowledge-aware recommender systems by evaluating their performance across both accuracy and fairness metrics. Using the MovieLens 1M dataset, we compare general, knowledge-aware, and fairness-optimized models through a custom RecBole-based pipeline. Results indicate knowledge-aware models offer some fairness benefits without major accuracy loss, though no model excels universally. Adjusting loss component weights reveals complex trade-offs and component importance, underscoring the need for nuanced fairness optimization.

1 Introduction

Recommender systems are tools which can help individuals and business stakeholders in their decision-making process. There are a few paradigms that have gained more popularity since this concept was first introduced - Collaborative filtering and Content-based filtering. Both aim to capture different aspects of available information, with the former being focused on the behaviour of similar users, while the latter utilizes the features of similar content, but the end goal is to help the user find desired content. In recent years, another paradigm called Knowledge-Aware Recommender Systems has gained momentum. This paradigm not only aims to provide natural explainability of results by exploiting facts from Knowledge Graphs, but related works have explored the possibility of combining this with other types of information such as embeddings derived from textual content, leading to improved accuracy [13].

However, bias in such recommendation systems remains a prominent concern brought about by already existing societal bias that is embedded in the data used for creating the system. It has been shown that Content-based and Collaborative filtering systems might exacerbate this bias problem creating unfair results such as underrepresentation of unpopular items or social minority groups [18]. Given that recommender systems are also used for social resource allocation, such as recommending jobs, this issue is not only related to the extent to which one perceives the accuracy of the system but also the ethical complications that come with using frameworks susceptible to bias [9]. One relevant study has shown that knowledge-aware models can still pick up on sensitive information embedded within the knowledge graphs fed to the model [17]. To that end, a new model was introduced that tries to take into account the complex mutual influence of sensitive attributes via a sensitivity graph, achieving state-of-the-art fairness results while also keeping the accuracy high.

Knowledge-aware systems are oftentimes evaluated on a set of widely used performance metrics such as MAE, RMSE, and NDCG [4] [22]. The main design choices of such systems are usually related to solving cold-start and data sparsity problems [22], with a large number of papers being focused on maximizing accuracy. What seems to be missing is a study that focuses on establishing the level of fairness of current state-of-the-art knowledge-based models. As such, three research questions will be explored:

- **RQ1:** Do knowledge-aware models perform better than other paradigms on metrics related to fairness?
- **RQ2:** Does adjusting the relative weights of components in the loss function of a knowledge-aware recommender system lead to improved performance on fairness and accuracy metrics?
- **RQ3:** Given the findings in RQ2, can optimizing for fairness, based on the selected metrics, affect the accuracy of the models in question? If so, to what extent?

Relevant metrics for accuracy and fairness based on existing literature will be utilized for testing. The test results will be compared pairwise between each model, and also benchmarked against the performance of baseline General Recommendation and Collaborative-Filtering models, as well as a state-of-the-art Fairness-aware model.

The rest of this paper is organized as follows: Section 2 presents the background; Section 3 presents the methodology alongside the experimental setup; Section 4 establishes the results derived from the pipeline; Section 5 is focused on a discussion about responsible research, followed by Section 6 which discusses the findings; Section 7 concludes the content of the paper and lays the groundwork for future research, while also acknowledging the current limitations.

2 Background

Recommender systems are a concept that was first introduced in the 90s [3]. Over time, their main use has come in the form of pruning the ever-increasing search space for users by providing them with content they would be interested in based on their behaviour. Nowadays, the most common classifications for such systems include content-based, collaborative, and knowledge-based techniques, with multiple methods being utilized in order to form the so-called hybrid systems [4]. Such systems have proven their worth in the industry, with companies such as Netflix and Amazon utilizing them for their movie and e-commerce services respectively [7][11]. With such systems being utilized by both individual users looking for leisure time content and by business stakeholders sifting through job applications [21], it is important to consider not only their accuracy, but also the ethical aspect of those systems, which can be expressed by estimating the fairness of their output. To that end, the scope of this paper revolves around assessing the fairness performance of current state-of-the-art knowledge-aware models and hybrid paradigms that include knowledge-based methods, all other systems are excluded from consideration.

2.1 Knowledge-Aware & Hybrid Recommendation

Graph-based and Knowledge-Graph Models

The idea of exploiting user-item graphs for recommendation has been researched for quite some time [12]. Early approaches utilized random walks to propagate user preferences, followed by Graph Convolutional Networks which were able to pick up on higher-order collaborative interactions based on the available graph information. Another type of a graph-based model is the knowledge graph based model. This model utilizes a knowledge graph, which represents higher-order semantic relationships between entities, rather than just simple user-item interactions.

Using knowledge graphs as side information in recommendation models can help diminish common limitations, such as data sparsity and the cold-start problem [8]. Evaluations seem to suggest that the precision in predictive power is improved for such systems. Moreover, using knowledge graphs as background information helps improve the explainability and trustworthiness of such systems. Advancements in deep learning techniques for graph data have given rise to new knowledge-aware, deep recommender systems based on Graph Neural Netwoks. There exists literature specifically on this topic [6] which focuses on leading frameworks, particularly the graph embedding modules they employ, and how they tackle key challenges such as scalability and cold-start problems.

Hybrid models

Over the years, as more research is done on each of the popular paradigms, it seems like problems such as the cold-start problem are becoming more ubiquitous, as no single paradigm has the innate capability of dealing with it completely. As all of the well-known recommendation techniques have strengths and weaknesses, researchers have shifted their focus to a more complicated paradigm - hybrid recommender systems. Such systems combine different recommendation methods in the hopes of achieving the best of both worlds while also dealing with the aforementioned problems.

When it comes to accuracy, there seem to be mixed results, with Burke [3] reporting benefits of hybrid recommendation with knowledge-aware components, while others conclude that even simple linear models can outperform such sophisticated paradigms [12]. As such, when it comes to the predictive power of hybrid models, no definitive conclusions can be derived, and it seems like there is no one-size-fits-all solution.

2.2 Fairness

Fairness in Recommender Systems has rapidly evolved into an important research area, driven by the significant societal and individual impact these systems wield. The current state reflects a field grappling with the complexity of defining fairness, which is inherently a multi-faceted, and often subjective social construct.

There's no single definition of fairness. Research explores various notions, including individual fairness (treating similar individuals similarly) and group fairness (ensuring equitable outcomes across demographic groups)[16][15]. Attention is also paid to user-side fairness (e.g., equitable recommendation quality across user groups like gender or age) and provider-side/item fairness (e.g., fair exposure for different items or item creators, often addressing popularity bias). The concepts of consistent fairness (similar treatment for similar entities) and calibrated fairness (outcomes proportional to merit) are frequently investigated [16][15][5].

The majority of research focuses on developing technical, algorithmic solutions. These typically fall into pre-processing (debiasing data), in-processing (modifying model training, e.g., via regularization or adversarial learning), or post-processing (re-ranking) strategies. Evaluation is predominantly conducted through offline experiments using historical datasets (e.g. MovieLens) and a variety of computational fairness metrics [5].

While process fairness (fairness of the recommendation model/process) is acknowledged, the bulk of research concentrates on achieving outcome fairness (fairness of the recommendation results) [16][15][5].

Fairness-Aware Models

Fairness-aware models predominantly aim to mitigate discriminatory outcomes by either modifying the learning algorithm's objective or transforming data representations. A common architectural pattern involves augmenting standard model training with regularization terms that penalize unfairness, as measured by specific metrics [19]. More sophisticated architectures leverage adversarial learning, setting up a minimax game between a primary model (e.g., a recommender) and an adversary that tries to predict sensitive attributes from the primary model's internal representations or outputs [10]. The objective of these adversarial setups is often to learn fair representations or embeddings that are useful for the main task but are invariant to protected attributes, preventing the model from relying on sensitive information, and achieving state-of-the-art fairness results [10].

3 Methodology

The main goal of this paper is to determine whether knowledge-aware models can provide fairer results when compared to baseline models and fairness-optimized models. In order to establish the current level of fairness for this paradigm, a pipeline was devised. This pipeline is a fork based on the Recbole 1.2.1 framework, combined with its FairRec Recbole 2.0 fairness derivation. This allows for reproducibility, as the datasets, models, and metrics described below are publicly accessible and integrated directly in the framework. This section aims to elaborate on the pipeline, which includes training the models, collecting and analyzing the results.

3.1 Datasets

The MovieLens 1M dataset was utilized for this study. It is a widely used benchmark dataset for evaluating recommender systems. It contains 1 million ratings (from 1 to 5 stars) collected from 6,000 users on 4,000 movies. The dataset includes demographic data such as gender, age, and occupation, but only gender was included as a sensitive attribute in this pipeline.

After the dataset was downloaded, it was configured to work for knowledge-aware models by creating knowledge graphs using the RecSysDatasets framework that is closely related to RecBole. Next, a 5-core filtering section was added to the pipeline, alongside duplicate removal, evaluation splits and field inclusion. All non-explicitly defined settings are automatically assigned default values according to the RecBole documentation. Given these settings, there were 6040 users in total, 4331 of which were male (71.7%) and 1709 were female (28.3%). Out of all interactions (997024 in total), 751192 belonged to the male group (75.3%), while 245832 belonged to the female group (24.7%).

3.2 Models

The models studied via this pipeline can be assigned to three different categories. Due to time constraints, a limited subset of models was picked for each category. Models were chosen based on popularity, suitability, and size of hyperparameter search space.

General Recommender Baselines

Three models made it into this category. These models are not specialized in fairness optimization, nor do they include or combine any sophisticated architecture.

- Popular - This is a non-personalized baseline model that simply recommends the most popular items (i.e., those with the highest number of interactions) to all users.
- Random - This model recommends items randomly, without considering user preferences or item popularity. It serves as a naive baseline to test the effectiveness of other models, especially when evaluating non-accuracy metrics like coverage or novelty.
- ItemKNN - This collaborative filtering model computes similarities between items and recommends items similar to those a user has interacted with [2].

Knowledge-aware models

Three models made it into this category. These models can be considered hybrid, as they utilize concepts from collaborative filtering and rely on knowledge graphs.

- CKE - This model integrates information from a knowledge base, such as textual, and visual data, into collaborative filtering by jointly learning item representations using embedding techniques which allows the model to improve recommendation performance in sparse settings by enriching item representations with semantic context from the knowledge graph [20].
- CFKG - This model enhances explainable recommendations by embedding users, items, and entities from a knowledge graph into a unified latent space, enabling personalized matching and explanation [1].
- RippleNet - This model simulates the process of user preference propagation on a knowledge graph by activating a series of "ripples" through multi-hop relations from a user's interacted items, thereby building a dynamic preference representation [14].

Fairness-aware models

Due to time constraints, only one model was chosen for this category. In contrast to the aforementioned models, it is the only one that provides a hyperparameter setting whose purpose is to optimize the model's fairness performance on a sensitive domain.

- PFCN-PMF - This model extends Probabilistic Matrix Factorization by inserting an adversarial "filter" module that removes user-sensitive attributes (e.g., gender, age) from learned embeddings before scoring. During training, it jointly minimizes the BPR ranking loss and maximizes the inability of per-attribute discriminators to predict those sensitive features from the filtered embeddings [10].

3.3 Metrics

The subset of metrics chosen for this study can be divided into three categories. All Top-K metrics were ran with K=10.

Accuracy

- Recall@10: Measures the fraction of relevant items that are successfully recommended.
- MRR@10 - Measures the rank of the first relevant item.
- NDCG@10 - Measures ranking quality, giving more weight to relevant items ranked higher.
- Hit@10 - Measures if at least one relevant item is in the top-K recommendations.
- MAP@10 - Averages precision at each recall point for relevant items.
- Precision@10 - Measures the fraction of recommended items that are relevant.
- GAUC (Grouped Area Under Curve) - Assesses recommendation quality by computing per-user discrimination between relevant and non-relevant items, then averaging those per-user scores weighted by each user's number of relevant interactions.

User-side Fairness

The following metrics are directly concerned with fairness, particularly user-side, group fairness between gender splits. Note that RecBole uses raw logits instead of predicted ratings, changing the result domain. As such, those logits were passed through a sigmoid function, after which they were linearly scaled to match the expected rating domain $[1, 5]$.

- Differential Fairness - This metric aims for equitable treatment across gender groups. It seeks to ensure that the recommendation outcome for an item is approximately the same regardless of the user's gender group.
- Value Unfairness - This measures the inconsistency in signed estimation error (i.e., whether the system consistently overestimates or underestimates) for items between different gender groups.
- Absolute Unfairness - This measures the inconsistency in the magnitude of estimation error (absolute error) for items between gender groups.

- Underestimation Unfairness - This metric quantifies the disparity in how much the system underestimates the true ratings for items between gender groups.
- Overestimation Unfairness - This quantifies the disparity in how much the system overestimates the true ratings for items between gender groups.
- Non-Parity Unfairness - This measures the absolute difference in the average predicted scores/ratings given to items by different gender groups.

Item-side fairness and diversity
These metrics, while not directly measuring disparity in outcomes between user gender groups, are often considered in fairness discussions as they relate to the diversity of recommended items

- Tail percentage @10 - Measures the proportion of recommended items that are from the "long-tail" (less popular items).
- Gini index - Measures the inequality in the distribution of recommended items (often based on item popularity). A lower Gini index indicates more diversity in recommended items, which can be seen as fairer to a wider range of item providers.
- Popularity percentage @10 - Measures the proportion of popular items in the recommendations.

3.4 Experimental Setup

As mentioned above, the pipeline mainly relies on Recbole 1.2.1, as well as the FairRec extension. The code for both was merged wherever necessary (e.g. metrics, class definitions, models definitions), the final version of which is available on GitHub [1].

Each of the aforementioned models was assigned a grid hyperparameter search space for a maximum of 150 epochs per permutation. The main validation metric was Recall@10, which determined the best models for later analytical use, as well as early stopping behaviour. Grid search was governed by the Weights & Biases platform, which helped with parallel CPU execution, as well as logging and data collection. For that reason, the Weights & Biases logger in RecBole was customized in order to accommodate for the logging behaviour and format.

During training, the evaluation dataset was used to create 2 more gender-filtered evaluation subsets - one for the male group, and another for the female group. To that end, metrics for 3 evaluation sets (All, Male, and Female) were calculated after each epoch, the main aim of which was to log metric progression for each of the user groups. After training was finished, all log data was pulled from the Weights & Biases platform via their API and ran through custom analysis scripts.

4 Results

This section presents the results derived from the experimental setup outlined in the previous section. Findings are grouped per research question

[1] https://github.com/JoanMVPopov/RecBole-tud-rp

4.1 RQ1

In order to summarize the evaluation results, Tables 1 and 2 were created. Table 1 provides the results for each model's best run, based on Recall@10, while Table 2 provides a mean z-score rank based on Table 1, aiming to take into account the models' performance across all metrics relevant to the their respective performance domain. For further insight, the data from the top 5 best performing runs for each model, based on Recall@10, was used to create Figure 1.

Accuracy
Overall accuracy, as detailed in Table 1, shows ItemKNN achieving the highest Recall@10 and NDCG@10, outperforming the knowledge-aware models, as well as all other models, in these accuracy aspects by a noticeable margin (further supported by its leading aggregate Z-score in Table 2). CFKG leads all models in gAUC (0.8711) and appears to perform the best among all knowledge-aware models in terms of Recall@10 (0.0763).

CKE demonstrates the best overall results among the knowledge-aware models when considering all accuracy metrics collectively via the Z-score aggregation (Table 2). The POP and PFCN-PMF models exhibit lower accuracy, with their overall performance being similar. The Random model performs the worst, which is also evident in its very low aggregate Z-score (Table 2). The gender-specific subtables (a and b below Table 1) reveal that the male group generally exhibits better best-run performance than the female group across most metrics, except for gAUC, where the female group achieved a score of 0.8740 with CFKG, compared to 0.8702 for the male group with the same model. Interestingly, the knowledge-aware models showed better best-run Recall@10 performance for the female group than for the male group. The boxplots in Figure 1 suggest that model performance varies more for the female group. However, it is possible, especially for knowledge-aware models, to choose a hyperparameter set that achieves better performance for that group.

User-side Fairness
The POP model achieves the highest aggregate Z-score (0.447, Table 2), indicating strong overall performance in this category. It notably scores best on Value Unfairness (0.5835, Table 1) and Absolute Unfairness (0.4804). The Random model also performs well in aggregate user-side fairness (Z-score 0.439). Among the knowledge-aware models, CFKG demonstrates the best user-side fairness profile, leading with the best scores for Differential Fairness (0.9398), Overestimation Unfairness (0.0037), and NonParity (0.0001), contributing to its positive aggregate Z-score (0.123). ItemKNN, despite its accuracy prowess, shows the poorest user-side fairness with the lowest aggregate Z-score (-0.595), exhibiting particularly undesirable values for Differential Fairness (1.5324) and Overestimation Unfairness (0.7547). CKE and RippleNet show intermediate performance, with RippleNet having a notably worse NonParity score (0.0256). Surprisingly, PFCN-PMF, the only fairness-aware model on the list, ranks right in the middle in terms of Z-score, being outperformed by a knowledge-aware paradigm (CFKG).

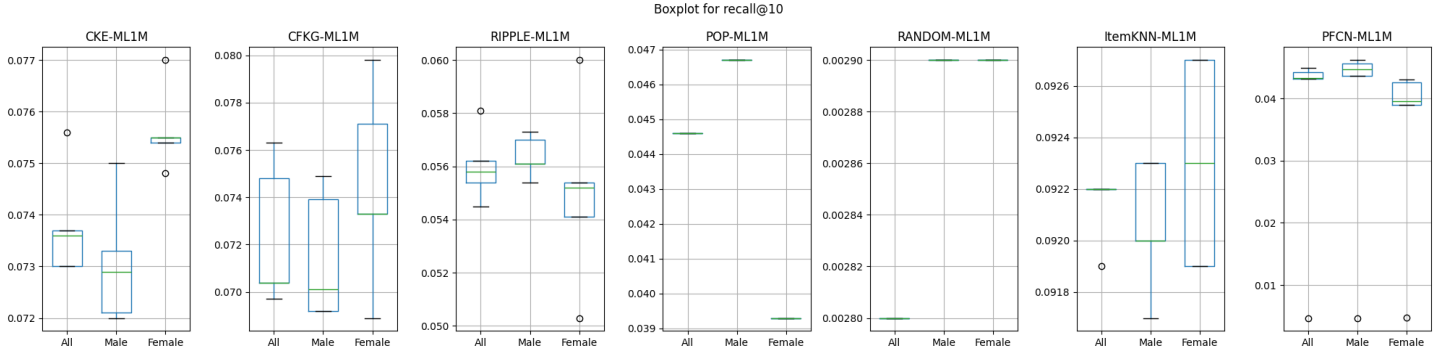


Figure 1: Boxplots for metric Recall@10, grouped by gender. The plots are based on the top 5 best-performing runs per model for this metric.

Table 1: Evaluation and fairness metrics results for best run per model, including gender-specific results at $K = 10$. Best runs are selected based on Recall@10. Arrows indicate whether higher (↑) or lower (↓) values are better.

	Accuracy (↑)							**User-side fairness (↓)**						**Item-side fairness & diversity**		
Model	Recall@10	MRR@10	NDCG@10	Hit@10	MAP@10	Precision@10	gAUC	Diff. Fair.	Value Unf.	Abs. Unf.	Underestim.	Overestim.	NonParity	Tail%@10↑	Gini@10↓	Pop%@10↓
CKE	0.0756	0.1633	0.0798	0.4639	0.0331	0.0608	0.8682	1.4013	0.7343	0.6147	0.3134	0.4209	0.0040	0.0003	0.9245	0.9166
CFKG	0.0763	0.1541	0.0769	0.4575	0.0314	0.0586	**0.8711**	**0.9398**	0.7765	0.7707	0.7728	**0.0037**	**0.0001**	0.0007	0.8709	0.8953
RippleNet	0.0581	0.1410	0.0659	0.4053	0.0263	0.0523	0.8543	1.3666	0.7642	0.6432	0.5180	0.2462	0.0256	0.0023	0.9544	0.9302
POP	0.0446	0.1231	0.0553	0.3490	0.0221	0.0442	0.8035	1.4358	**0.5835**	**0.4804**	0.2363	0.3472	0.0095	0.0000	0.9971	1.0000
Random	0.0032	0.0130	0.0052	0.0444	0.0017	0.0047	0.5020	1.4565	0.6049	0.5059	0.2176	0.3873	0.0021	**0.2015**	**0.1341**	**0.1967**
ItemKNN	**0.0922**	**0.2058**	**0.1012**	**0.5281**	**0.0456**	**0.0725**	0.8602	1.5324	0.7728	0.7569	**0.0180**	0.7547	0.0102	0.0001	0.9540	0.9765
PFCN-PMF	0.0449	0.1241	0.0559	0.3464	0.0225	0.0443	0.8005	1.4483	0.5887	0.4875	0.2132	0.3755	0.0319	0.0000	0.9960	1.0000

Gender-specific evaluation metrics at $K = 10$ (best runs).

(a) Female group

	Accuracy (↑)		**Item-side fairness & diversity**		
Model	Recall@10	gAUC	Tail%@10↑	Gini@10↓	Pop%@10↓
CKE	0.0770	0.8685	0.0001	0.9311	0.8978
CFKG	0.0798	**0.8740**	0.0005	0.8849	0.8728
RippleNet	0.0600	0.8519	0.0023	0.9557	0.9147
POP	0.0393	0.7945	0.0000	0.9971	1.0000
Random	0.0029	0.4996	**0.1954**	**0.2538**	**0.2040**
ItemKNN	**0.0919**	0.8593	0.0000	0.9524	0.9741
PFCN	0.0431	0.7960	0.0000	0.9970	1.0000

(b) Male group

	Accuracy (↑)		**Item-side fairness & diversity**		
Model	Recall@10	gAUC	Tail%@10↑	Gini@10↓	Pop%@10↓
CKE	0.0750	0.8682	0.0003	0.9296	0.9241
CFKG	0.0749	**0.8702**	0.0008	0.8797	0.9041
RippleNet	0.0573	0.8551	0.0023	0.9579	0.9363
POP	0.0467	0.8064	0.0000	0.9971	1.0000
Random	0.0029	0.4997	**0.1980**	**0.1618**	**0.2003**
ItemKNN	**0.0923**	0.8605	0.0001	0.9581	0.9775
PFCN	0.0456	0.8020	0.0000	0.9966	1.0000

Table 2: Z-score normalized aggregate scores across metric groups. Z-score normalization was applied on Table 1, per column. The means for each model's z-score were calculated per row, inverting the z-score result wherever necessary, such that the final aggregated score implies "higher is better".

Accuracy Metrics

Model	Z-score
ItemKNN	1.205
CKE	0.617
CFKG	0.544
RippleNet	0.183
PFCN	-0.203
POP	-0.209
Random	-2.137

User Fairness Metrics

Model	Z-score
POP	0.447
Random	0.439
CFKG	0.123
PFCN	0.078
CKE	-0.048
RippleNet	-0.444
ItemKNN	-0.595

Item Fairness Metrics

Model	Z-score
Random	2.433
CFKG	-0.242
CKE	-0.332
RippleNet	-0.374
ItemKNN	-0.442
PFCN	-0.520
POP	-0.522

Item-side Fairness and Diversity

Examining item-side fairness and diversity, the Random model overwhelmingly demonstrates the best performance in this category, achieving an exceptionally high aggregate Z-score of 2.433. This is driven by its outstanding scores in Tail Percentage (0.2015), Gini Index (0.1341), and Popularity Percentage (0.1967), indicating it recommends a highly diverse and non-popular set of items. Among the other models, CFKG shows the best item-side fairness profile with the second-highest aggregate Z-score (-0.242, though considerably lower than Random), primarily due to having the lowest (best) Gini Index (0.8709) and Popularity Percentage (0.8953) among the non-Random models, alongside a reasonable Tail Percentage (0.0007). CKE and RippleNet follow, with RippleNet exhibiting the highest Tail Percentage (0.0023) among the non-Random models, suggesting better long-tail coverage. However, RippleNet's Gini Index (0.9544) and Popularity Percentage (0.9302) are less favorable compared to CFKG and CKE. ItemKNN, POP, and PFCN-PMF perform poorly in this category, with POP and PFCN-PMF showing no tail coverage (Tail% = 0.0) and very high Gini Index and Popularity Percentage scores, indicating a strong bias towards popular items. ItemKNN also struggles with a low Tail Percentage (0.0001) and high Gini (0.9540) and Popularity Percentage (0.9765).

4.2 RQ2

The knowledge-aware models RippleNet and CKE were investigated further, as their loss function naturally consists of a recommendation loss component and a KG loss component. The model CFKG was not considered as its loss function definition did not provide a natural split for those components. Two additional hyperparameters were included, α_{rec} and α_{kg}, which were responsible for assigning weight to each component. For α_{rec}, the values were drawn from the list [0.5, 1.0, 1.5, 2.0], while for α_{kg} that list was [0.0, 0.5, 1.0, 1.5, 2.0]. Both models were trained anew with the same early stopping and metric maximization mechanism as in RQ1. Due to time constraints, the models' respective hyperparameters, including the newly introduced ones, were chosen from the top five best-performing runs (Figure 1) that also offered reasonable training time - not necessarily the best, but among the top and fastest to train.

Due to this setup, α_{rec} and α_{kg} can be defined as independent variables, and all other metrics can be defined as dependent. This is a suitable configuration for the Two-way ANOVA test, which was carried out on those variables in order to detect significance. The variables were treated as categorical for the purposes of the formula that's fed into the ANOVA model, not only to not assume linearity, but also due to the restricted domain for those values (as defined above). Tukey's HSD tests were conducted on significant results in order to establish the direction of significance, the baselines for which were defined beforehand, based on proximity to the respective default RecBole values (CKE: $\alpha_{kg} = 1.0$; RippleNet: $\alpha_{kg} = 0.0$; both: $\alpha_{rec} = 1.0$). Statistical significance was determined at $p < 0.05$.

Accuracy

The CKE baseline generally maintained strong performance. Specifically, for the 'male' user group, the CKE baseline $\alpha_{kg} = 1.0$ was significantly better than $\alpha_{kg} < 1.0$ for Hit@10, Precision@10, and Recall@10. For all groups, having $\alpha_{kg} > 1.0$ led to slightly better results, but not in a significant way. However, while the male group experienced a significant decrease in Recall@10 for lower α_{kg} values, the female group showed an increase for $\alpha_{kg} = 0$. For RippleNet, the baseline $\alpha_{kg} = 0.0$ was often more effective for accuracy with 'male' users (NDCG, Precision@10, Recall@10 being better than in a non-zero α_{kg} configuration). Across both models, variations in α_{rec} from its 1.0 baseline did not yield significant improvements in accuracy metrics. Overall, alternative hyperparameter settings did not offer broad, statistically significant enhancements in core accuracy metrics beyond the established baselines for either model.

User-Side Fairness

In terms of user-side fairness for CKE, α_{kg} values higher than its baseline demonstrated a slight but non-significant improvement across the metrics. However, setting $\alpha_{kg} = 0$ yielded contradictory results - Underestimation Unfairness was significantly worse, but Differential, Non-Parity, and Overestimation Unfairness exhibited nearly significant improvements. For RippleNet, the results indicated that non-zero α_{kg} yielded slightly better Overestimation Unfairness values. However, such values also led to significantly worse results for almost all other metrics except for Non-Parity Unfairness, where no significance was detected. Changes in the α_{rec} parameter did not lead to significant changes in user-side fairness for either model.

Item-Side Fairness and Diversity

For item-side fairness diversity, the CKE model's baseline did not see significant changes from alternative settings. However, RippleNet demonstrated notable benefits when α_{kg} was adjusted from its 0.0 baseline. Specifically, setting $\alpha_{kg} = 1.5$ led to a significantly better Tail Percentage for 'all', 'female', and 'male' user groups. Furthermore, for the 'female' segment, $\alpha_{kg} = 1.5$ also achieved a significantly better Gini Index, while also being on the brink of significance for the other user groups, which was also the case for Popularity Percentage. Changes to α_{rec} did not yield changes improvements in this category for either model.

4.3 RQ3

The fairness–accuracy trade-off was assessed using two 2D multiobjective scatterplots per model (Figure 2), with a min-maxed Recall@10 on one axis and a composite fairness score on the other. For each fairness side, the appropriate metrics were min–max-scaled (inverting wherever necessary) and then averaged out per model in order to produce an aggregate score. This score was further min-max-scaled in order to produce proper plots and calculations for euclidean distance to the optimal point.

Looking at the frontier plots for CKE (plots a and b from Figure 2), a trend is clearly visible - as we increase the Recall@10 metric, the composite fairness metric seems to get slightly worse. However, this tradeoff might be worth investigating. For user-fairness, picking the point that has the shortest euclidean distance to the optimal point yields 17.13% average improvement per user-side fairness metric whilst giving up only 2.04% in terms of Recall@10 performance when compared with the base configuration (Table 3). Similar results can be seen for item-fairness, with an 18.28% increase on per fairness metric on average and a 3.12% decrease for Recall@10 (Table 4).

As far as RippleNet is concerned, the same general tradeoff trend between Recall@10 and composite user-side fairness can be seen (Figure 2, subplots c and d). However, it seems like the base configuration is amongst the most optimal configurations for user-side fairness, with the point with shortest euclidean distance only yielding a 0.36% improvement for Recall@10 and a slight 0.03% decrease per fairness metric (Table 3). In contrast, an interesting result can be observed for the item-side composite fairness metric where picking the most optimal point for item-side fairness yields an impressive 80.63% average increase whilst only reducing the Recall@10 performance by 0.72% (Table 4).

Table 3: User-side fairness and Recall@10 changes

Model	Avg. user-side fairness change	Recall@10 change
CKE	17.13%	-2.04%
RippleNet	-0.03%	0.36%

5 Responsible Research

The data used in this study is publicly accessible. Only a limited set of sensitive attributes is retained for training and analysis, that attribute being gender in this study.

Table 4: Item-side fairness and Recall@10 changes

Model	Avg. item-side fairness change	Recall@10 change
CKE	18.28%	-3.12%
RippleNet	80.63%	-0.72%

The training and evaluation of the recommender models were conducted using an open-source framework. A specific fork of this framework, customized for the purposes of this study, is also publicly available. Additionally, all scripts used for data preprocessing, model evaluation, and metric computation are included in the repository linked in this study. The figures presented in this paper are generated directly from those scripts and are also available in the repository.

To ensure reproducibility, all software dependencies and experimental configurations are explicitly defined. Given an identical setup, the results reported in this study should be replicable by other researchers.

6 Discussion

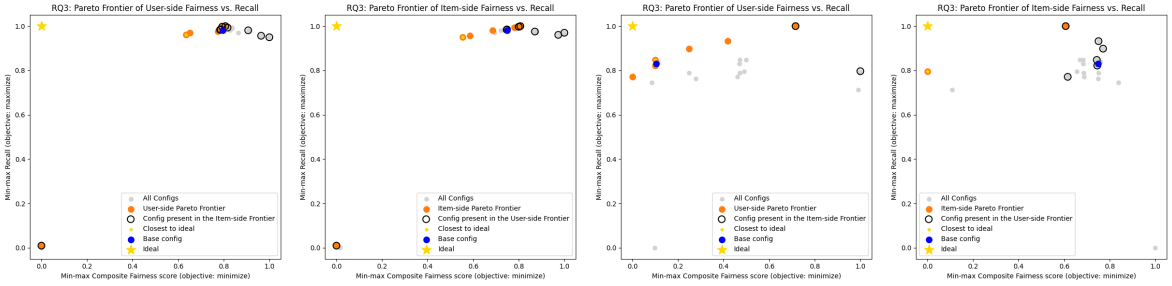
The findings in Section 4 reveal a complex interplay between model architecture, hyperparameter tuning, and the multifaceted nature of fairness.

6.1 "Jack of all trades, master of none"

The comparative analysis in RQ1 underscores a key takeaway, based on the investigated dataset: no single model or paradigm emerged as a universal champion across all performance domains. While the collaborative filtering model ItemKNN dominated pure accuracy metrics, its performance on user-side fairness was notably poor. Conversely, simpler baselines like POP and Random excelled in specific fairness domains – POP in user-side fairness and Random in item-side diversity – albeit at the cost of accuracy. Knowledge-aware models occupied an interesting middle ground. CFKG demonstrated the best overall user-side fairness among similar models and even surpassed the fairness-aware PFCN-PMF model in this aggregate category. CKE, while strong in aggregate accuracy, did not particularly stand out for metrics from any fairness side, but was not a poor performer either. RippleNet offered competitive item-side diversity, but user-side fairness was not its strong suit. This pattern, where a knowledge-aware model consistently ranked within the top three in aggregate Z-scores for accuracy (CFKG, CKE), user-side fairness (CFKG), and item-side fairness (CFKG, CKE), suggests their potential for achieving a more holistic fairness performance while retaining competitive accuracy.

6.2 The Critical Role of Knowledge Graph Weighting

The investigation into loss component weighting (RQ2) for CKE and RippleNet revealed the significant leverage provided by the knowledge graph component weight (α_{kg}), while the recommendation loss weight (α_{rec}) showed minimal impact within the tested range. For CKE, its baseline proved robust for accuracy, particularly for male users, and deviating to $\alpha_{kg} = 0.0$ significantly worsened Underestimation Unfairness for



(a) CKE - User-side fairness (b) CKE - Item-side fairness (c) RippleNet - User-side fairness (d) RippleNet - Item-side fairness

Figure 2: Scatterplots displaying the Pareto frontiers for CKE and RippleNet

all users. This indicates that the KG component is integral to CKE's performance and fairness profile. RippleNet's behavior was more varied: its baseline was often better for male user accuracy and some user-side fairness aspects, yet increasing α_{kg} to 1.5 yielded substantial improvements in item-side diversity across all user groups, while also significantly reducing user-side fairness performance. This illustrates that optimizing for one fairness aspect or user group via α_{kg} may inadvertently affect others, necessitating careful, context-specific tuning. It may possible to further develop the loss functions of those models by making the separate loss component weights learnable, adjusting them based on internally calculated fairness and accuracy metrics.

6.3 Navigating the Fairness-Accuracy Frontier

The exploration of the fairness–accuracy trade-off (RQ3) confirmed its existence but also highlighted opportunities for optimization. For CKE, enhancing either user-side or item-side composite fairness was achievable with a relatively small recall decrease compared to its baseline configuration. This suggests that meaningful fairness gains can be made without crippling accuracy. RippleNet presented an even more compelling case for item-side fairness: an impressive 80.63% average improvement in composite item-side fairness was possible with a negligible 0.72% reduction in recall. While its user-side fairness showed less room for improvement over the baseline via this method, the potential to significantly boost item diversity with minimal accuracy cost is a valuable finding. These results demonstrate that the trade-off is not always severe and that carefully selected hyperparameter configurations can lead to models that are both reasonably accurate and demonstrably fairer. It is also worth noting that there seems to be a tradeoff between user-side fairness and item-side fairness. While there was some overlap between the frontiers, the majority of the points, which belonged to one fairness domain's frontier, did not belong to the frontier of the other fairness domain. This implies that optimizing for both user-side and item-side fairness is a non-trivial task, the interdependence relationships in which need to be studied further.

7 Conclusions and Future Work

This research investigated the complex landscape of fairness in knowledge-aware recommender systems, comparing their performance against general and fairness-focused models, and examining the impact of internal component weighting. Our findings reveal that no single model universally excels across accuracy and all fairness dimensions. While traditional models like ItemKNN lead in accuracy, they often falter in user-side fairness. Conversely, simpler baselines can achieve strong results in specific fairness domains but at a significant accuracy cost. Knowledge-aware models, such as CFKG and CKE, demonstrate a promising ability to achieve a more holistic performance, often ranking competitively across accuracy, user-side, and item-side fairness domains without dominating any single one.

The weighting of the knowledge graph component in models like CKE and RippleNet proved to be a significant lever for tuning, capable of substantially impacting both accuracy and various fairness metrics, sometimes with contradictory effects across different fairness aspects or user groups. The recommendation component weight showed less influence. Furthermore, the exploration of the fairness-accuracy trade-off indicated that substantial gains in specific fairness dimensions, particularly item-side diversity for RippleNet and balanced improvements for CKE, are often achievable with only minor compromises in accuracy. These results highlight the potential for targeted optimization. However, different types of inherent trade-offs emphasize the need for nuanced, well-defined strategies to create recommender systems that are both truly fair and effective.

Limitations and Future Work

Despite the valuable findings, this study has several limitations that may affect the generalizability and interpretability of the results:

- Experimental Scope - The study used a limited number of datasets and models, with fixed group splits and a single 8/1/1 train/validation/test split. Broader evaluations with more datasets, diverse group definitions, more model types, and multiple splits would enhance the robustness of the findings.
- Hyperparameter and Metric Coverage - The hyperparameter search space (e.g., for α_{rec} and

α_{kg}) was relatively narrow, which may have missed fine-grained performance-fairness trade-offs. Similarly, the set of fairness metrics was not exhaustive, limiting the comprehensiveness of the fairness evaluation. Those need to be expanded - including more well-defined fairness metrics calculated across a larger search space.

- No ubiquitous framework - Because of the current state of the RecBole framework, some fairness metric implementations used in this study do not exactly match canonical definitions in the literature, which may limit comparability with prior work. More work needs to be done in order to ensure easier and more reproducible pipeline setups when using such open-source projects.

References

[1] Qingyao Ai, Vahid Azizi, Xu Chen, and Yongfeng Zhang. Learning heterogeneous knowledge base embeddings for explainable recommendation. *Algorithms*, 11(9):137, 2018.

[2] Fabio Aiolli. Efficient top-n recommendation for very large scale binary rated datasets. In *Proceedings of the 7th ACM Conference on Recommender Systems (RecSys '13)*, pages 273–280. ACM, 2013.

[3] Robin Burke. Hybrid recommender systems: Survey and experiments. *User Modeling and User-Adapted Interaction*, 12(4):331–370, 2002.

[4] Viktoriia Danilova and Andrew Ponomarev. Hybrid recommender systems: The review of state-of-the-art research and applications. 01 2016.

[5] Yashar Deldjoo, Dietmar Jannach, Alejandro Bellogín, et al. Fairness in recommender systems: Research landscape and future directions. *User Modeling and User-Adapted Interaction*, 34(1):59–108, 2024.

[6] Yang Gao, Yi-Fan Li, Yu Lin, Hang Gao, and Latifur Khan. Deep learning on knowledge graph for recommender system: A survey, 2020.

[7] Carlos A. Gomez-Uribe and Neil Hunt. The netflix recommender system. *ACM Transactions on Management Information Systems*, 6(4):13:1–13:19, December 2015.

[8] Andreea Iana, Mehwish Alam, and Heiko Paulheim. A survey on knowledge-aware news recommender systems. *Semantic Web*, 15(1):21–82, 2024.

[9] Deepak Kumar, Tessa Grosz, Navid Rekabsaz, Elisabeth Greif, and Markus Schedl. Fairness of recommender systems in the recruitment domain: an analysis from technical and legal perspectives. *Frontiers in Big Data*, Volume 6 - 2023, 2023.

[10] Yunqi Li, Hanxiong Chen, Shuyuan Xu, Yingqiang Ge, and Yongfeng Zhang. Towards personalized fairness based on causal notion. In *SIGIR 2021 - Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR 2021 - Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, pages 1054–1063. Association for Computing Machinery, Inc, July 2021. Publisher Copyright: © 2021 ACM.; 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2021 ; Conference date: 11-07-2021 Through 15-07-2021.

[11] Greg Linden, Brent Russell Smith, and Nida K. Zada. Collaborative Recommendations Using Item-to-Item Similarity Mappings, June 2001. Issued to Amazon.com.

[12] Qingsong Lv, Ming Ding, Qiang Liu, Yuxiang Chen, Wenzheng Feng, Siming He, Chang Zhou, Jianguo Jiang, Yuxiao Dong, and Jie Tang. Are we really making much progress? revisiting, benchmarking and refining heterogeneous graph neural networks. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, KDD '21, page 1150–1160, New York, NY, USA, 2021. Association for Computing Machinery.

[13] Giuseppe Spillo, Cataldo Musto, Marco Polignano, Pasquale Lops, Marco de Gemmis, and Giovanni Semeraro. Combining graph neural networks and sentence encoders for knowledge-aware recommendations. In *Proceedings of the 31st ACM Conference on User Modeling, Adaptation and Personalization*, UMAP '23, page 1–12, New York, NY, USA, 2023. Association for Computing Machinery.

[14] Hongwei Wang, Fuzheng Zhang, Jialin Wang, Miao Zhao, Wenjie Li, Xing Xie, and Minyi Guo. Ripplenet: Propagating user preferences on the knowledge graph for recommender systems. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, CIKM '18, page 417–426, New York, NY, USA, 2018. Association for Computing Machinery.

[15] Yifan Wang, Weizhi Ma, Min Zhang, Yiqun Liu, and Shaoping Ma. A survey on the fairness of recommender systems. *ACM Trans. Inf. Syst.*, 41(3), February 2023.

[16] Yao Wu, Jian Cao, and Guandong Xu. Fairness in recommender systems: Evaluation approaches and assurance strategies. *ACM Trans. Knowl. Discov. Data*, 18(1), August 2023.

[17] Bingke Xu, Yue Cui, Zipeng Sun, Liwei Deng, and Kai Zheng. Fair representation learning in knowledge-aware recommendation. In *2021 IEEE International Conference on Big Knowledge (ICBK)*, pages 385–392, 2021.

[18] Emre Yalcin and Alper Bilge. Evaluating unfairness of popularity bias in recommender systems: A comprehensive user-centric analysis. *Information Processing Management*, 59(6):103100, 2022.

[19] Sirui Yao and Bert Huang. Beyond parity: Fairness objectives for collaborative filtering. In

I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[20] F. Zhang, N. J. Yuan, D. Lian, X. Xie, and W.-Y. Ma. Collaborative knowledge base embedding for recommender systems. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '16)*, pages 353–362. ACM, August 2016.

[21] Shuo Zhang and Peter J Kuhn. Measuring bias in job recommender systems: Auditing the algorithms. Working Paper 32889, National Bureau of Economic Research, August 2024.

[22] Erion Çano and Maurizio Morisio. Hybrid recommender systems: A systematic literature review. *Intelligent Data Analysis*, 21(6):1487–1524, 2017.