Figure 5: *Precision@K*, *Recall@K*, and *F1@K* in top-*K* recommendation for MovieLens-1M.

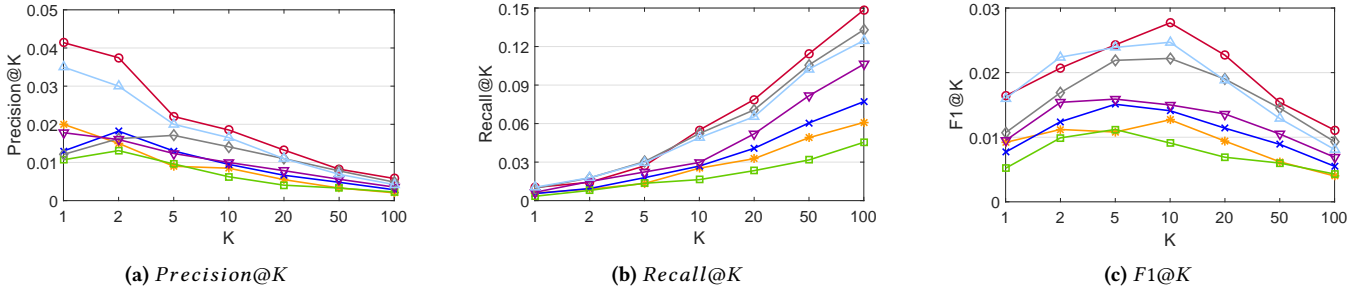**Figure 6:** *Precision@K*, *Recall@K*, and *F1@K* in top-*K* recommendation for Book-Crossing.
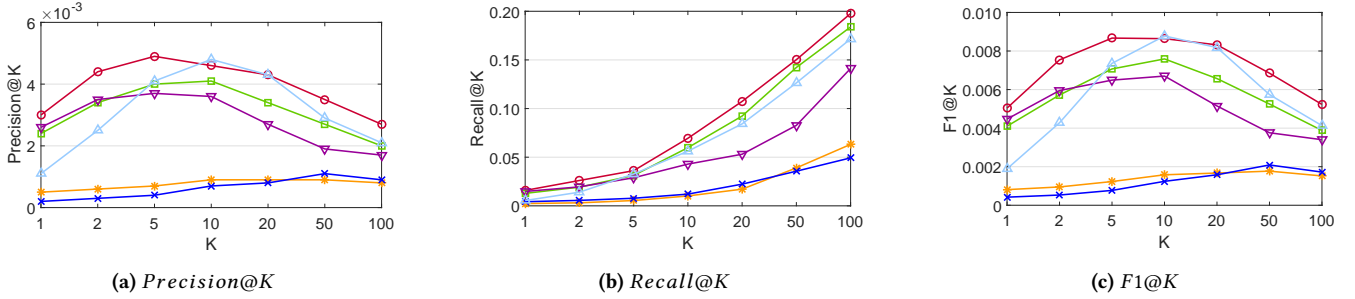

Figure 7: *Precision@K*, *Recall@K*, and *F1@K* in top-*K* recommendation for Bing-News.

Table 3: The results of *AUC* and *Accuracy* in CTR prediction.

Model	MovieLens-1M		Book-Crossing		Bing-News	
	AUC	*ACC*	*AUC*	*ACC*	*AUC*	*ACC*
RippleNet*	**0.921**	**0.844**	**0.729**	**0.662**	**0.678**	**0.632**
CKE	0.796	0.739	0.674	0.635	0.560	0.517
SHINE	0.778	0.732	0.668	0.631	0.554	0.537
DKN	0.655	0.589	0.621	0.598	0.661	0.604
PER	0.712	0.667	0.623	0.588	-	-
LibFM	0.892	0.812	0.685	0.639	0.644	0.588
Wide&Deep	0.903	0.822	0.711	0.623	0.654	0.595

* Statistically significant improvements by unpaired two-sample *t*-test with $p = 0.1$.

- DKN performs best in news recommendation compared with other baselines, but performs worst in movie and book recommendation. This is because movie and book names are too short and ambiguous to provide useful information.
- PER performs unsatisfactorily on movie and book recommendation because the user-defined meta-paths can hardly

be optimal. In addition, it cannot be applied in news recommendation since the types of entities and relations involved in news are too complicated to pre-define meta-paths.
- As two generic recommendation tools, LibFM and Wide&Deep achieve satisfactory performance, demonstrating that they can make well use of knowledge from KG into their algorithms.
- RippleNet performs best among all methods in the three datasets. Specifically, RippleNet outperforms baselines by 2.0% to 40.6%, 2.5% to 17.4%, and 2.6% to 22.4% on *AUC* in movie, book, and news recommendation, respectively. RippleNet also achieves outstanding performance in top-*K* recommendation as shown in Figures 5, 6, and 7. Note that the performance of top-*K* recommendation is much lower for Bing-News because the number of news is significantly larger than movies and books.

Size of ripple set in each hop. We vary the size of a user's ripple set in each hop to further investigate the robustness of RippleNet. The results of *AUC* on the three datasets are presented in Table 4,