Tiny-NewsRec: Effective and Efficient PLM-based News Recommendation

Yang Yu, Fangzhao Wu, Chuhan Wu, Jingwei Yi and Qi Liu

Microsoft Research
微软亚洲研究院

Email: yflyl613@mail.ustc.edu.cn
Code: https://github.com/yflyl613/Tiny-NewsRec/

Introduction

Background

- News recommendation is widely used to improve user experience.
- Learning high-quality news representations from news texts is one of the most critical tasks for news recommendation.
- Pre-trained language models (PLMs) have benefited news recommendation by improving news modeling.

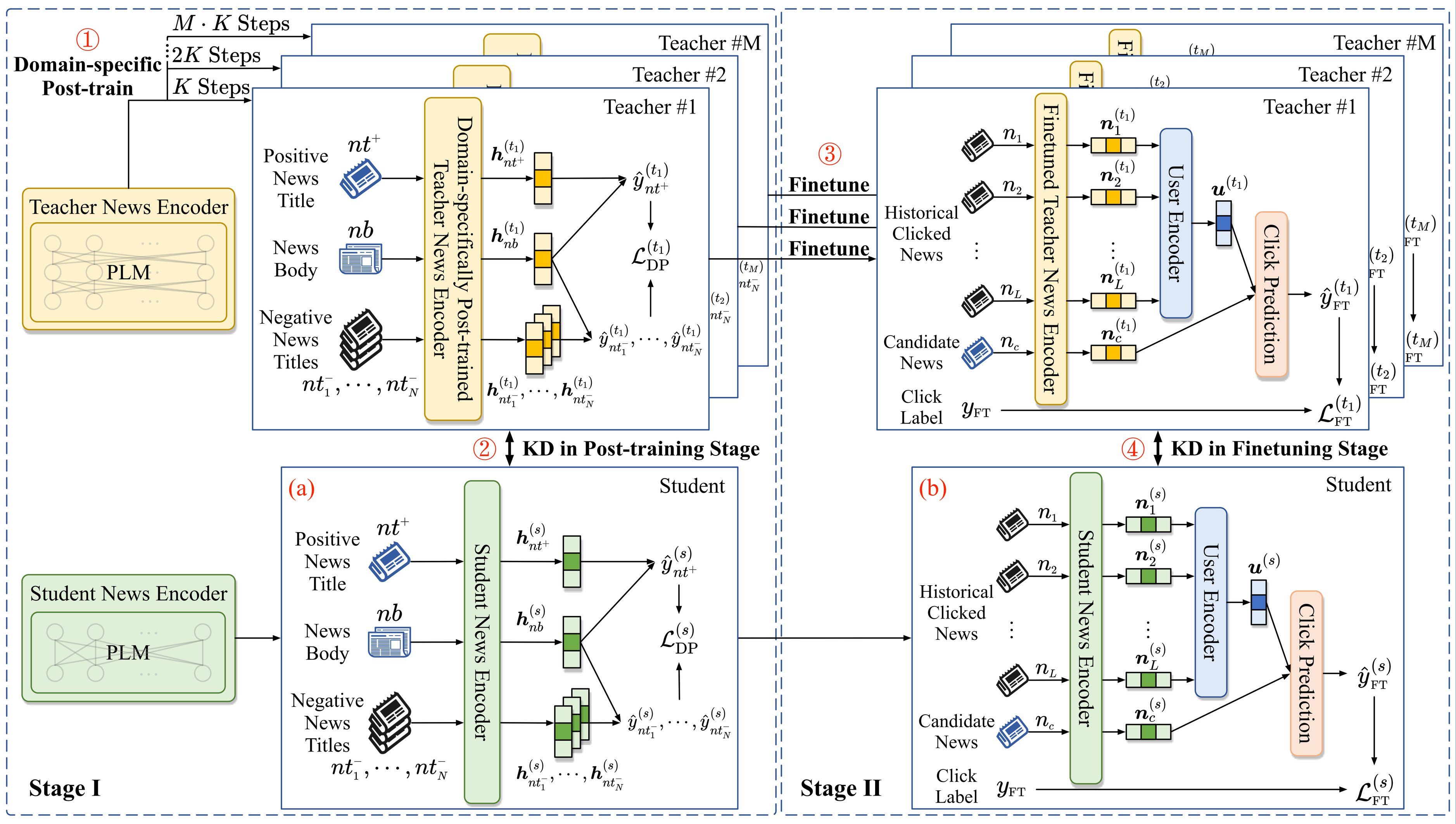
Challenges

- Simply finetuning the general PLM with the news recommendation task may suffer from the domain shift problem.
- Deploying large PLM-based news recommendation models for online services requires extensive computational resources.

Tiny-NewsRec

Main Idea

- Adapt the general PLM to the news domain with self-supervised domain-specific post-training before the task-specific finetuning.
- Compress the large PLM-based model with a two-stage knowledge distillation method.



Domain-specific Post-training

- Different parts of a news article are naturally related.
- Train the PLM-based news encoder with a self-supervised contrastive matching task between news titles and news bodies.

$$\mathcal{L}_{\mathrm{DP}} = -\log \frac{\exp(\boldsymbol{h}_{nb}^{\mathrm{T}}\boldsymbol{h}_{nt^+})}{\exp(\boldsymbol{h}_{nb}^{\mathrm{T}}\boldsymbol{h}_{nt^+}) + \sum_{i=1}^{N}\exp(\boldsymbol{h}_{nb}^{\mathrm{T}}\boldsymbol{h}_{nt_i^-})}$$

nb: news body
nt^+: corresponding news title
nt_i^-: randomly sampled news title

Two-stage Knowledge Distillation

- Step 1: Post-train the teacher news encoder. A copy is saved every K steps and we save M teacher models in total.
- Step 2: Transfer domain-specific knowledge from these teachers to the student model during its post-training.

$$\alpha^{(t_i)} = \frac{\exp(-\mathrm{CE}(\hat{\boldsymbol{y}}_{\mathrm{DP}}^{(t_i)}, y_{\mathrm{DP}}))}{\sum_{j=1}^{M}\exp(-\mathrm{CE}(\hat{\boldsymbol{y}}_{\mathrm{DP}}^{(t_j)}, y_{\mathrm{DP}}))}$$

the adaptive weight of each teacher model on each sample

$$\mathcal{L}_{\mathrm{DP}}^{\mathrm{distill}} = T_{\mathrm{DP}}^2 \cdot \mathrm{CE}(\sum_{i=1}^{M}\alpha^{(t_i)}\hat{\boldsymbol{y}}_{\mathrm{DP}}^{(t_i)}/T_{\mathrm{DP}}, \hat{\boldsymbol{y}}_{\mathrm{DP}}^{(s)}/T_{\mathrm{DP}})$$

$$\mathcal{L}_{\mathrm{DP}}^{\mathrm{emb}} = \sum_{i=1}^{M}\alpha^{(t_i)}[\mathrm{MSE}(\boldsymbol{W}^{(t_i)}\boldsymbol{h}_{nt}^{(t_i)} + \boldsymbol{b}^{(t_i)}, \boldsymbol{h}_{nt}^{(s)}) + \mathrm{MSE}(\boldsymbol{W}^{(t_i)}\boldsymbol{h}_{nb}^{(t_i)} + \boldsymbol{b}^{(t_i)}, \boldsymbol{h}_{nb}^{(s)})]$$

$$\mathcal{L}_1 = \mathcal{L}_{\mathrm{DP}}^{\mathrm{distill}} + \mathcal{L}_{\mathrm{DP}}^{\mathrm{emb}} + \mathcal{L}_{\mathrm{DP}}^{(s)}$$

the overall loss function for the student model in Stage I

- Step 3: Finetune these M teacher models with the news recommendation task.
- Step 4: Transfer task-specific knowledge from these teachers to the student model during its finetuning.

$$\beta^{(t_i)} = \frac{\exp(-\mathrm{CE}(\hat{\boldsymbol{y}}_{\mathrm{FT}}^{(t_i)}, y_{\mathrm{FT}}))}{\sum_{j=1}^{M}\exp(-\mathrm{CE}(\hat{\boldsymbol{y}}_{\mathrm{FT}}^{(t_j)}, y_{\mathrm{FT}}))}$$

the adaptive weight of each teacher model on each sample

$$\mathcal{L}_{\mathrm{FT}}^{\mathrm{distill}} = T_{\mathrm{FT}}^2 \cdot \mathrm{CE}(\sum_{i=1}^{M}\beta^{(t_i)}\hat{\boldsymbol{y}}_{\mathrm{FT}}^{(t_i)}/T_{\mathrm{FT}}, \hat{\boldsymbol{y}}_{\mathrm{FT}}^{(s)}/T_{\mathrm{FT}}) \qquad \mathcal{L}_{\mathrm{FT}}^{(s)} = \mathrm{CE}(\hat{\boldsymbol{y}}_{\mathrm{FT}}^{(s)}, y_{\mathrm{FT}})$$

$$\mathcal{L}_{\mathrm{FT}}^{\mathrm{emb}} = \sum_{i=1}^{M}\beta^{(t_i)}[\mathrm{MSE}(\boldsymbol{W}_n^{(t_i)}\boldsymbol{n}^{(t_i)} + \boldsymbol{b}_n^{(t_i)}, \boldsymbol{n}^{(s)}) + \mathrm{MSE}(\boldsymbol{W}_u^{(t_i)}\boldsymbol{u}^{(t_i)} + \boldsymbol{b}_u^{(t_i)}, \boldsymbol{u}^{(s)})]$$

$$\mathcal{L}_2 = \mathcal{L}_{\mathrm{FT}}^{\mathrm{distill}} + \mathcal{L}_{\mathrm{FT}}^{\mathrm{emb}} + \mathcal{L}_{\mathrm{FT}}^{(s)}$$

the overall loss function for the student model in Stage II

Experiments

Datasets

- *MIND*: a public news recommendation dataset.
- *Feeds*: a news recommendation dataset collected on the MSN App.
- *News*: news articles collected on the MSN website.

MIND			
# News	161,013	# Users	1,000,000
# Impressions	15,777,377	# Clicks	24,155,470
Avg. title length	11.52		
Feeds			
# News	377,296	# Users	10,000
# Impressions	320,925	# Clicks	437,072
Avg. title length	11.93		
News			
# News	1,975,767	Avg. title length	11.84
Avg. body length	511.43		

Detailed statistics of *MIND*, *Feeds*, and *News*.

Performance Comparison

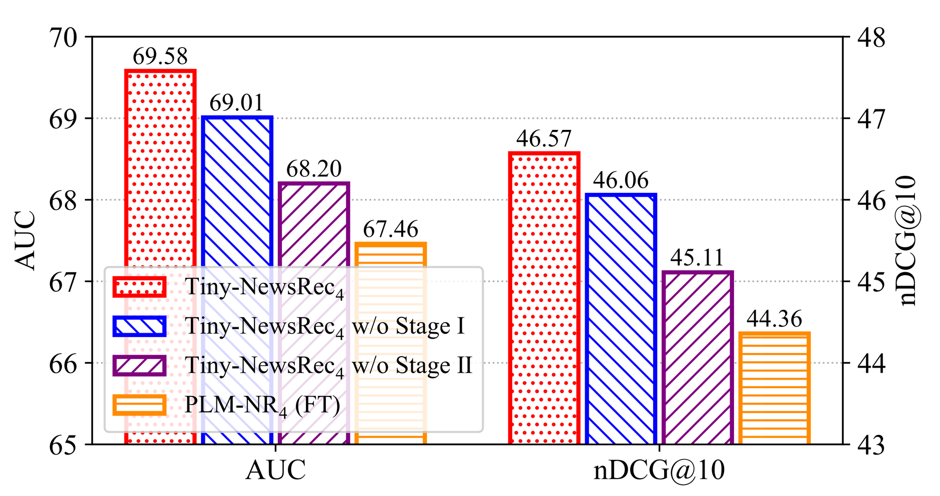
Model	MIND			Feeds			Model
	AUC	MRR	nDCG@10	AUC	MRR	nDCG@10	Size
PLM-NR$_{12}$ (FT)	69.72±0.15	34.74±0.10	43.71±0.07	67.93±0.13	34.42±0.07	45.09±0.07	109.89M
PLM-NR$_{12}$ (DAPT)	69.97±0.08	35.07±0.15	43.98±0.10	68.24±0.09	34.63±0.10	45.30±0.09	109.89M
PLM-NR$_{12}$ (TAPT)	69.82±0.14	34.90±0.11	43.83±0.07	68.11±0.11	34.49±0.12	45.11±0.08	109.89M
PLM-NR$_{12}$ (DP)	**71.02±0.07**	**36.05±0.09**	**45.03±0.12**	**69.37±0.10**	**35.74±0.11**	**46.45±0.11**	109.89M
PLM-NR$_4$ (FT)	69.49±0.14	34.40±0.10	43.40±0.09	67.46±0.12	33.71±0.11	44.36±0.09	53.18M
PLM-NR$_2$ (FT)	68.99±0.08	33.59±0.14	42.61±0.11	67.05±0.14	33.33±0.09	43.90±0.12	39.01M
PLM-NR$_1$ (FT)	68.12±0.12	33.20±0.07	42.07±0.10	66.26±0.10	32.55±0.12	42.99±0.09	31.92M
TinyBERT$_4$	70.55±0.10	35.60±0.12	44.47±0.08	68.40±0.08	34.64±0.10	45.21±0.11	53.18M
TinyBERT$_2$	70.24±0.13	34.93±0.07	43.98±0.10	68.01±0.07	34.37±0.09	44.90±0.10	39.01M
TinyBERT$_1$	69.19±0.09	34.35±0.10	43.12±0.07	67.16±0.11	33.42±0.07	43.95±0.07	31.92M
NewsBERT$_4$	70.62±0.15	35.72±0.11	44.65±0.08	68.69±0.10	34.90±0.08	45.64±0.11	53.18M
NewsBERT$_2$	70.41±0.09	35.46±0.07	44.35±0.10	68.24±0.08	34.64±0.11	45.23±0.10	39.01M
NewsBERT$_1$	69.45±0.11	34.75±0.09	43.54±0.12	67.37±0.05	33.55±0.10	44.12±0.08	31.92M
Tiny-NewsRec$_4$	**71.19±0.08**	**36.21±0.05**	**45.20±0.09**	**69.58±0.06**	**35.90±0.11**	**46.57±0.07**	53.18M
Tiny-NewsRec$_2$	70.95±0.04	36.05±0.08	44.93±0.10	69.25±0.07	35.45±0.09	46.25±0.10	39.01M
Tiny-NewsRec$_1$	70.04±0.06	35.16±0.10	44.10±0.08	68.31±0.03	34.65±0.08	45.32±0.08	31.92M

Performance of different methods on *MIND* and *Feeds*.

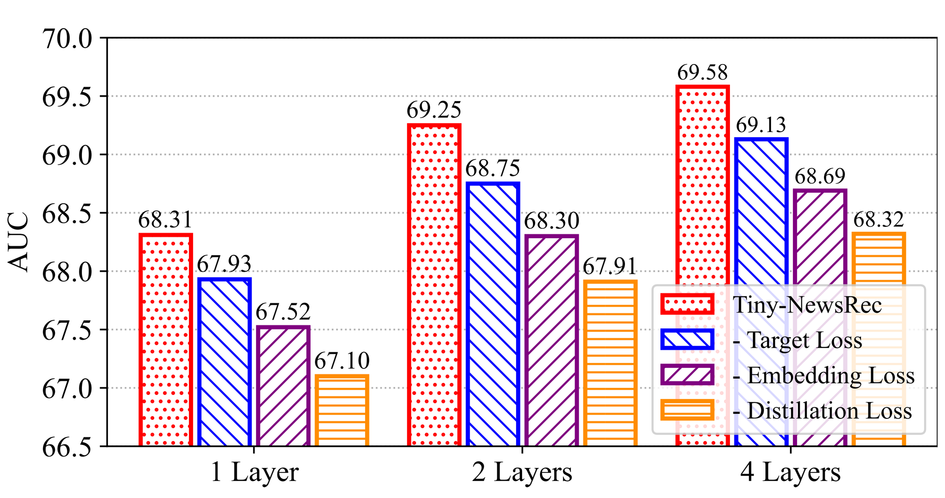
Model	AUC	MRR	nDCG@10
Ensemble-Teacher$_{12}$	69.43	35.81	46.53
TinyBERT-MT$_4$	68.87	35.13	45.81
NewsBERT-MT$_4$	68.82	35.07	45.80
MT-BERT$_4$	68.51	34.74	45.45
Tiny-NewsRec$_4$	**69.58**	**35.90**	**46.57**

Performance of different methods with multiple teacher models on *Feeds*.

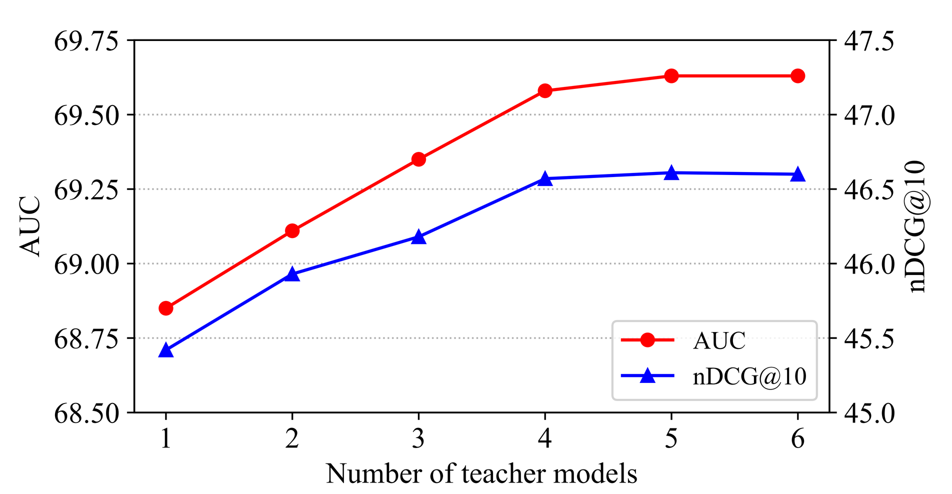
Ablation Study



Effectiveness of each stage.

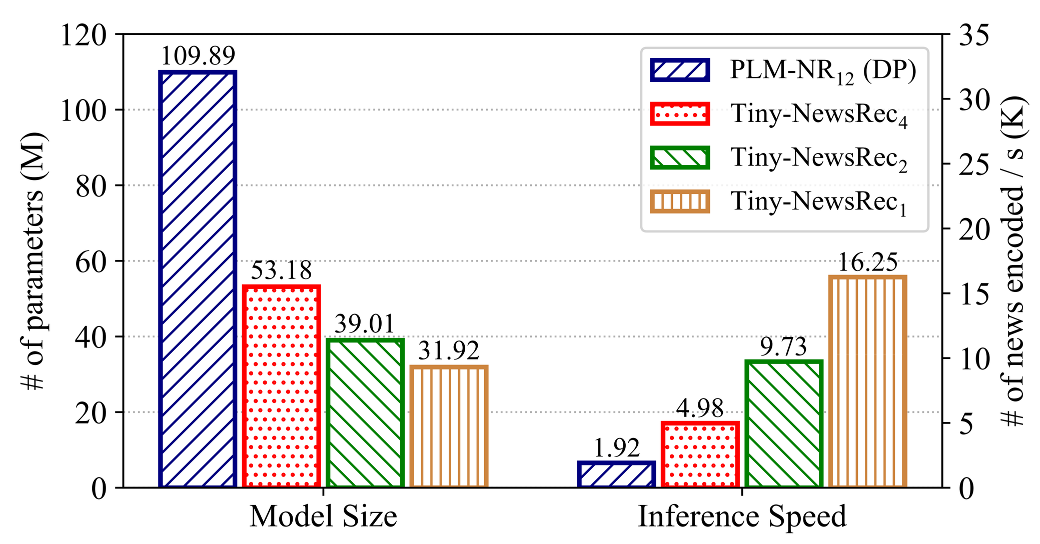


Effectiveness of each loss function.



Impact of the number of teacher models.

Efficiency Evaluation



Model size and inference speed of different models.