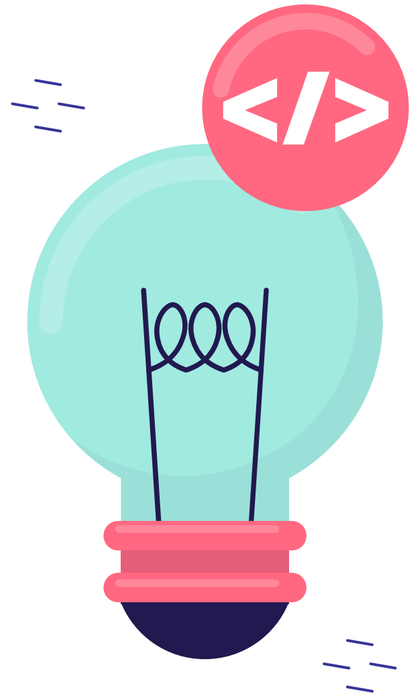
Tiny-NewsRec: Effective and Efficient PLM-based News Recommendation

Yang Yu, Fangzhao Wu, ChuhanWu, Jingwei Yi & Qi Liu

Grupo 10: Felipe Abarca, Alfredo Enrione, Nicolás Estévez

TABLA DE CONTENIDOS



01
CONTEXTO

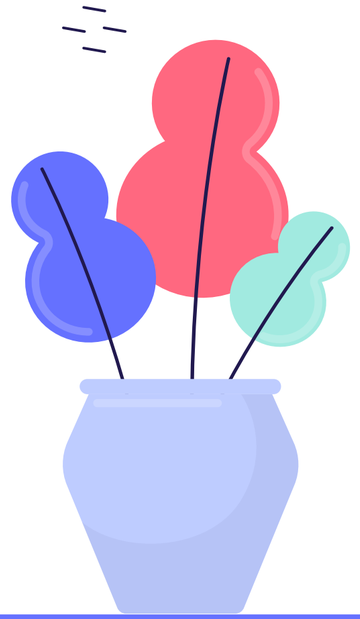
Contexto

La cantidad de informacion en internet se ha masificado de manera exponencial

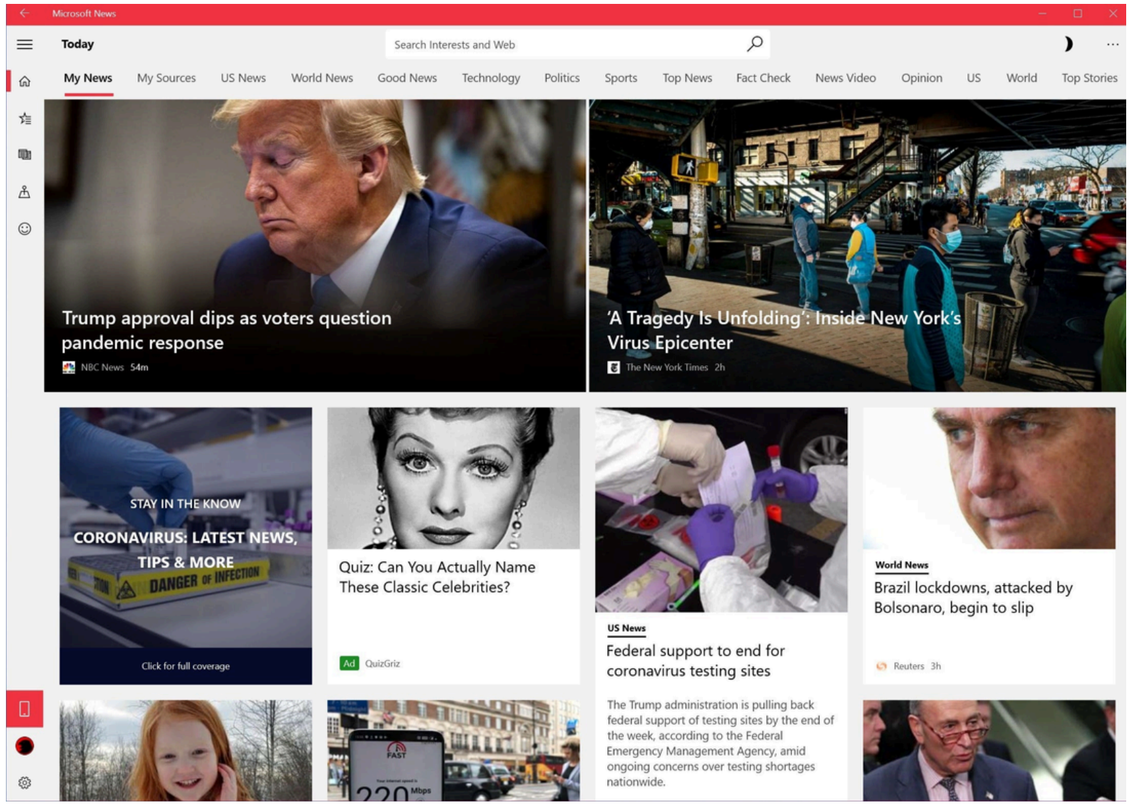
La sobre carga de informacion dificulta la toma de decisiones para los usuarios

En el area de recomendaciones, se ha visto popularidad en el enfoque en el procesamiento del lenguaje natural (NLP):

- Modelos costosos de entrenar computacionalmente
- No usan en su totalidad metodos enfocados en noticias sino en lenguaje generico



La importancia de las noticias



emol.

02

PROBLEMA DE RECOMENDACIÓN Y CONTRIBUCIÓN

Tiny-NewsRec

Problema de recomendación:

- Recomendar las noticias de preferencia del usuario
- Considerar partes relevantes de la noticia (título, cuerpo, palabras claves, etc.)
- Basado en el historial de noticias vistas por el usuario

Contribución:

- Mejor desempeño que modelos actuales
- Reducción de costos computacionales sin perjudicar su desempeno
- Mejorar tiempo de respuesta para predicciones
- Proporcionar un modelo que se entrena en el dominio de las noticias

¿Cómo mejorar la eficiencia y efectividad del sistema de recomendación de noticias ya existentes basado en PLMs?

03

ESTADO DEL ARTE Y MARCO TEÓRICO

Marco teórico

- **PLM:** Modelos de lenguaje pre entrenados sobre lenguaje natural, ej: BERT, UniLMv2

- **Knowleadge Distillation (KD):** Técnica de compresión de modelos, donde un modelo estudiante aprende del maestro (teacher-student, multiple teachers)

- **Contrastive Matching:** maximizar la información mutua entre partes semánticamente relacionadas de un texto (título y cuerpo)

Estado del Arte

- **PLM-NR (Wu et al., 2021b):** Hace fintetuning con clicks de usuario, no comprime el modelo de lenguaje
- **TinyBERT (Jiao et al., 2020):** usa KD para compresión del modelo pero usa BERT sin modificarlo
- **NewsBERT (Wu et al., 2021c):** usa KD pero se hace finetuning con datasets de noticias

Limitaciones

Sufren de una brecha entre los datos entrenados y lo que se quiere recomendar (domain-gap)

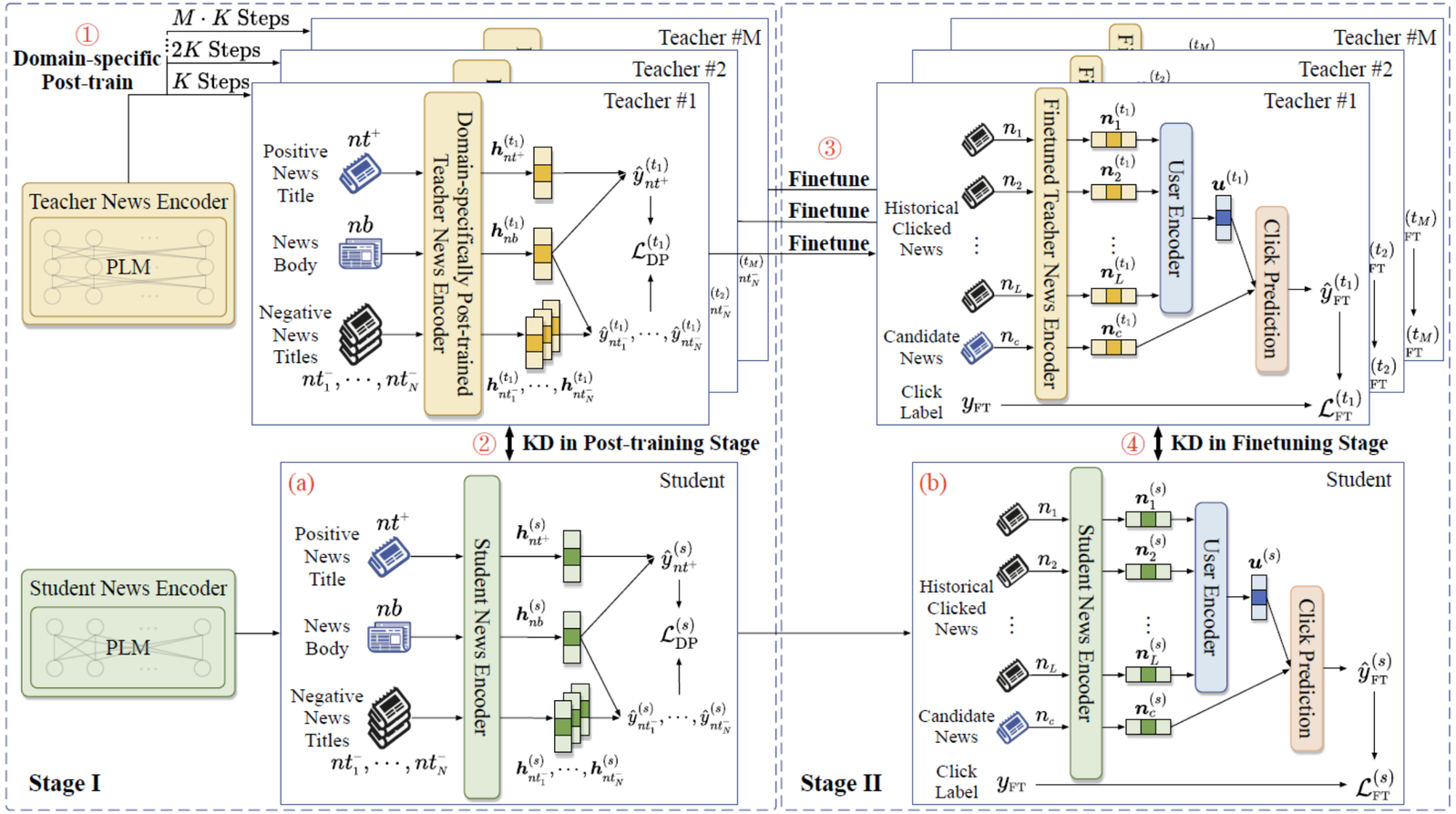
Contienen gran numero de parametros y son costosos de entrenar

Uso de 1 solo teacher no garantiza resultado optimo

04

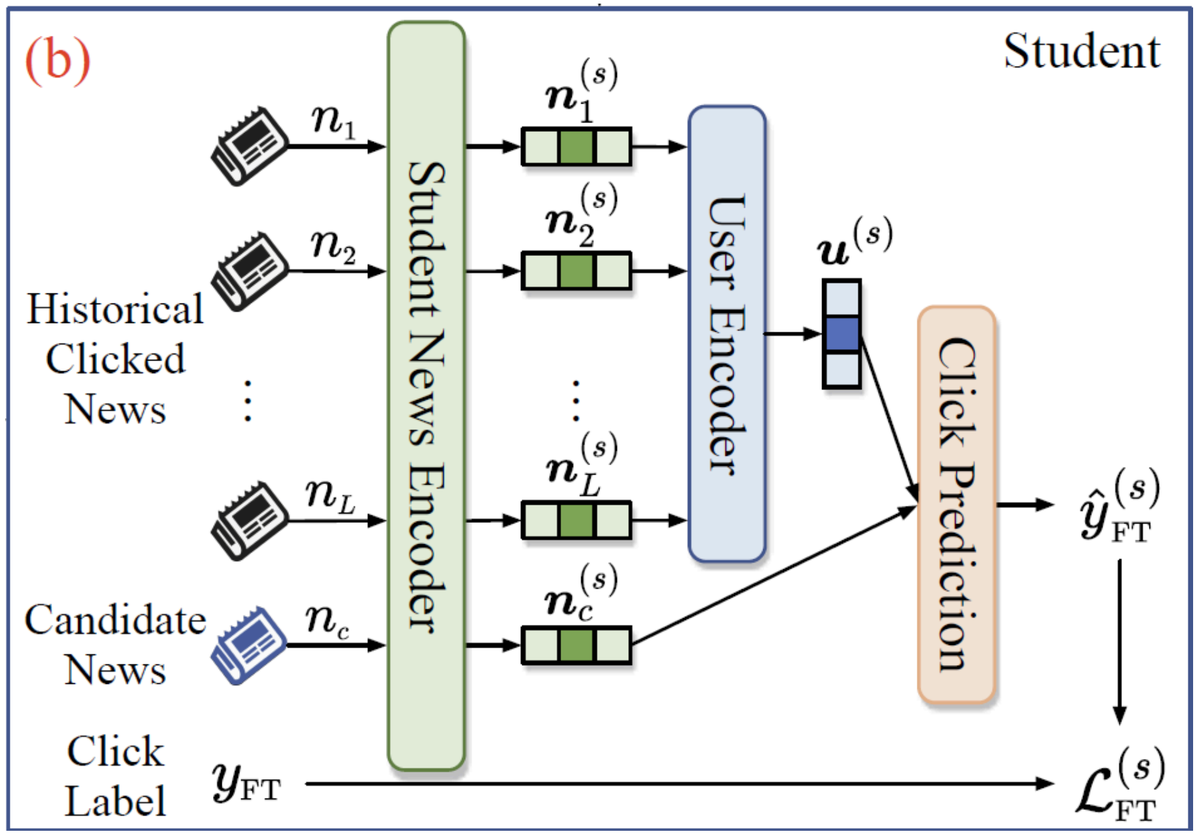
DETALLE SOLUCIÓN

Tiny-NewsRec



PLM-based news recommendation

$$\mathcal{L}_{FT}^{(s)} = CE(\hat{\boldsymbol{y}}_{FT}^{(s)}, y_{FT})$$



Domain-Specific Post-Training

¿Qué es?

Etapa de ajuste diseñada para adaptar los PLMs al dominio particular de las noticias. En lugar de entrenar desde cero o usar directamente el modelo general, este proceso permite que el PLM comprenda mejor las características lingüísticas, semánticas y estructurales propias de los textos periodísticos.

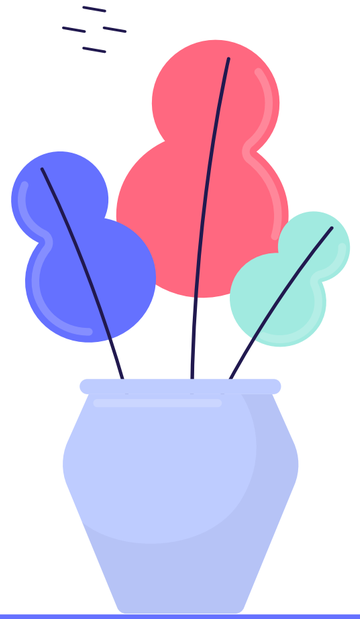
¿Cómo funciona?

Se utiliza una tarea auto-supervisada basada en contrastive matching entre los títulos y los cuerpos de las noticias. El modelo es entrenado para maximizar la similitud entre el cuerpo y su título correcto, y minimizarla con respecto a los títulos no relacionados.

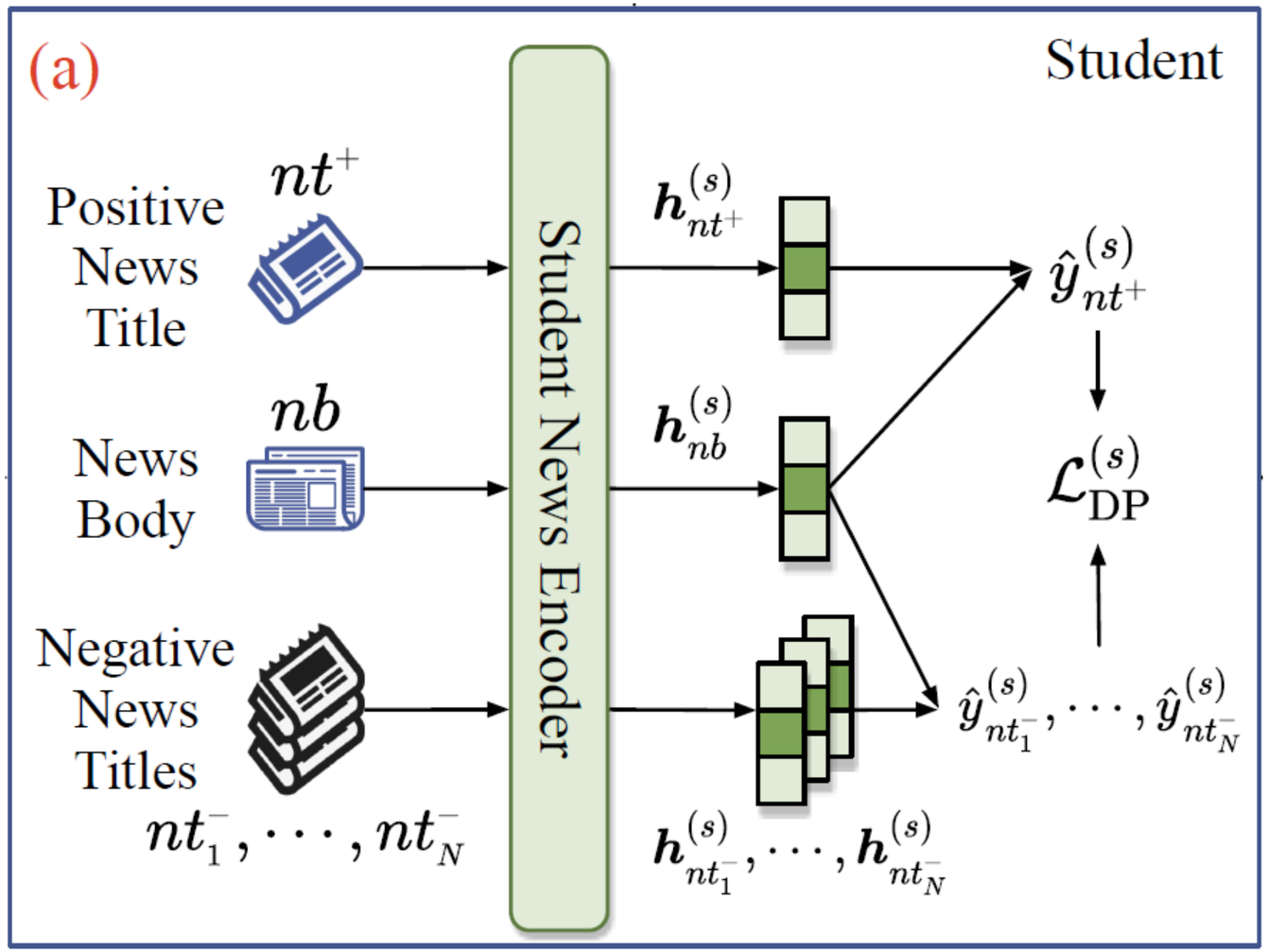
Función de pérdida

$$\mathcal{L}_{DP} = -\log \frac{\exp(\hat{y}_{nt^+})}{\exp(\hat{y}_{nt^+}) + \sum_{i=1}^{N} \exp(\hat{y}_{nt_i^-})}$$

$$\hat{y}_{nt^+} = \boldsymbol{h}_{nb}^T \boldsymbol{h}_{nt^+} \qquad \hat{y}_{nt^-} = \boldsymbol{h}_{nb}^T \boldsymbol{h}_{nt_i^-}$$



Domain-Specific Post-Training



Two-stage Knowledge Distillation

Stage I

Guardado teacher

Durante el entrenamiento se guardan copias de los modelos luego de K pasos, y se guardan M teacher

Funciones de pérdida

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

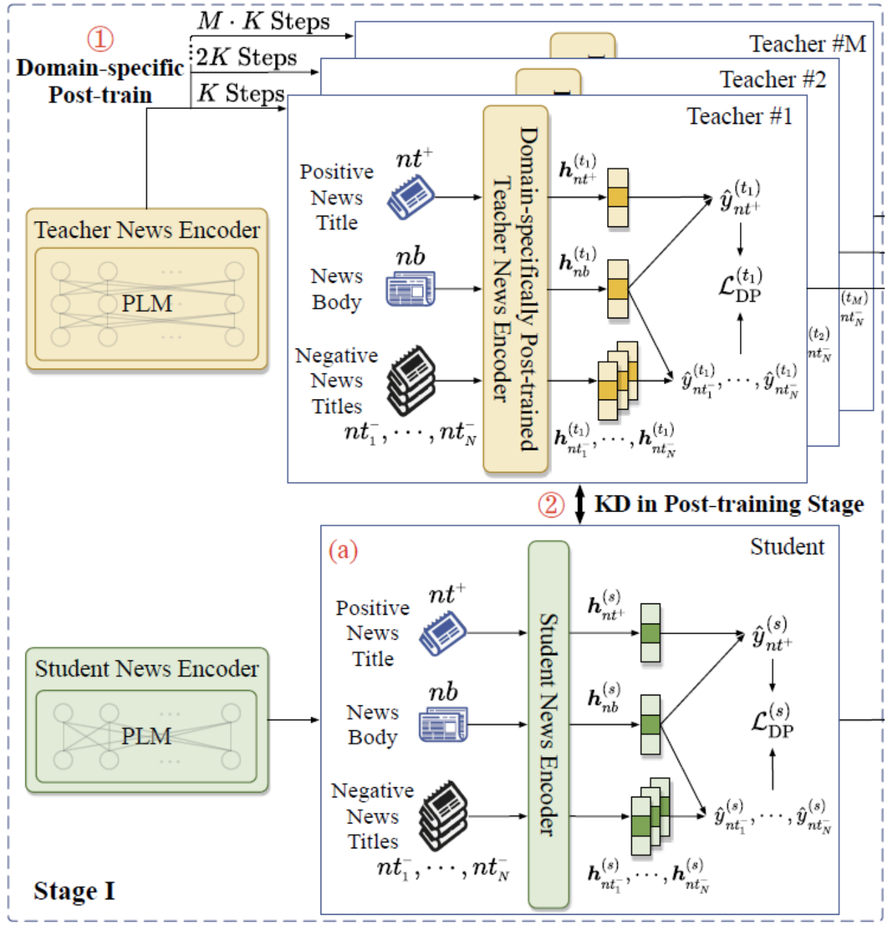
$$\mathcal{L}_{\text{DP}}^{\text{distill}} = T_{\text{DP}}^2 \cdot \text{CE}\left(\sum_{i=1}^{M} \alpha^{(t_i)} \hat{\boldsymbol{y}}_{\text{DP}}^{(t_i)} / T_{\text{DP}},\ \hat{\boldsymbol{y}}_{\text{DP}}^{(s)} / T_{\text{DP}}\right)$$

$$\mathcal{L}_{\text{DP}}^{\text{emb}} = \sum_{i=1}^{M} \alpha^{(t_i)} \mathcal{L}_{\text{DP}}^{\text{emb}_i} \quad \alpha^{(t_i)} = \frac{\exp\left(-\text{CE}(\hat{\boldsymbol{y}}_{\text{DP}}^{(t_i)}, y_{\text{DP}})\right)}{\sum_{j=1}^{M} \exp\left(-\text{CE}(\hat{\boldsymbol{y}}_{\text{DP}}^{(t_j)}, y_{\text{DP}})\right)}$$

$$\mathcal{L}_{\text{DP}}^{\text{emb}_i} = \text{MSE}(\boldsymbol{W}^{(t_i)} \boldsymbol{h}_{nt}^{(t_i)} + \boldsymbol{b}^{(t_i)}, \boldsymbol{h}_{nt}^{(s)}) + \text{MSE}(\boldsymbol{W}^{(t_i)} \boldsymbol{h}_{nb}^{(t_i)} + \boldsymbol{b}^{(t_i)}, \boldsymbol{h}_{nb}^{(s)})$$

Two-stage Knowledge Distillation

Stage I



Two-stage Knowledge Distillation

Stage II

Guardado teacher

Se le realiza fine-tuning a los M teachers del paso anterior. E igualmente, se ocupa una función de pérdida

Funciones de pérdida

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

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

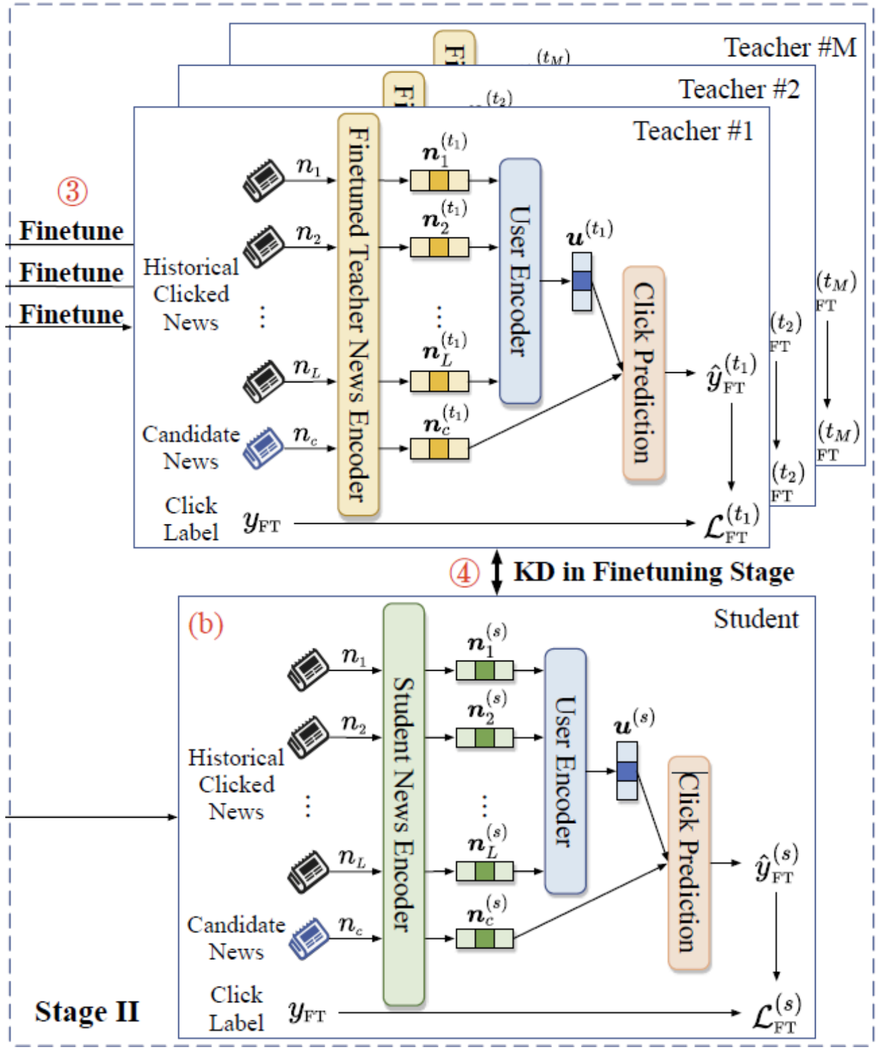
$$\beta^{(t_i)} = \frac{\exp\left(-\text{CE}(\hat{\boldsymbol{y}}_{\text{FT}}^{(t_i)}, y_{\text{FT}})\right)}{\sum_{j=1}^{M} \exp\left(-\text{CE}(\hat{\boldsymbol{y}}_{\text{FT}}^{(t_j)}, y_{\text{FT}})\right)}$$

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

$$\mathcal{L}_{\text{FT}}^{(s)} = \text{CE}(\hat{\boldsymbol{y}}_{\text{FT}}^{(s)}, y_{FT})$$

Two-stage Knowledge Distillation

Stage II



05

EVALUACIÓN

Evaluación

Datasets

- **MIND**
  - 1M usuarios de Microsoft News
  - 6 semanas de logs de clics
- **Feeds**
  - Microsoft News App
  - 1 mes de clics (ago–sep 2020)
  - 20% de training set para validación
- **News**
  - Noticias de Microsoft News (sep–oct 2020)
  - Usado para el domain-specific post-training

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		

Evaluación

Montaje Experimental

- **PLM base:** UniLMv2
- **Dimensiones:** 256 (usuario y noticia)
- **Nº de teachers:** 4 (copia de teacher cada 500 pasos)
- **Optimización:** Adam
- **Repeticiones:** Cada experimento se repite 5 veces
- **Métricas de evaluación**
  - AUC – Discriminación entre relevantes/no relevantes
  - MRR – Ranking de la primera noticia relevante
  - nDCG@10 – Calidad del orden en las 10 primeras

Evaluación

Resultados

- Presentados en 2 partes:

COMPARACIÓN DE PERFORMANCE

EVALUACIÓN COMPLEMENTARIA

Comparación de Performance

En esta sección se compara el rendimiento del modelo teacher PLM-NR_{12} (DP) y los modelos student entrenados con Tiny-NewsRec (1, 2 y 4 capas), con los siguientes métodos baselines:

PLM-NR (FT)
- Método base state-of-the-art.
- Fine-tuning directo del PLM (12 capas y versiones reducidas: 1, 2 y 4 capas).

PLM-NR (DAPT)
- Preentrenamiento adicional con textos del dominio de noticias (Domain-Adaptive Pretraining) antes del fine-tuning.

PLM-NR (TAPT)
- Preentrenamiento adicional adaptado a la tarea específica de recomendación (Task-Adaptive Pretraining), luego fine-tuning.

TinyBERT
- Método state-of-the-art de destilación en dos etapas (pretraining + fine-tuning).
- Usa como teacher a PLM-NR12 (con post-training de dominio).

NewsBERT
- Método state-of-the-art con destilación especializada en noticias.
- Entrena en conjunto teacher y student durante el fine-tuning.

Comparación de Performance

Model	MIND			Feeds			Model Size
	AUC	MRR	nDCG@10	AUC	MRR	nDCG@10	
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±0.09	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

Comparación de Performance

Comparación Adicional

Para entender mejor el origen de las mejoras de Tiny-NewsRec, se compara con los siguientes métodos que también utilizan múltiples teacher:

Ensemble-Teacher

- Conjunto de los modelos de 12 capas utilizados por Tiny-NewsRec.
- Para la evaluación se utiliza la puntuación media prevista de estos modelos teacher.

TinyBERT-MT / NewsBERT-MT

- Versiones modificadas de TinyBERT y NewsBERT que usan los múltiples teachers utilizados por Tiny-NewsRec.
- Cada teacher se pondera según su rendimiento en la muestra de entrenamiento de entrada (igual que en Tiny-NewsRec).

MT-BERT

- Modelo que entrena conjuntamente el student y múltiples teachers con distintos PLMs.

Comparación de Performance

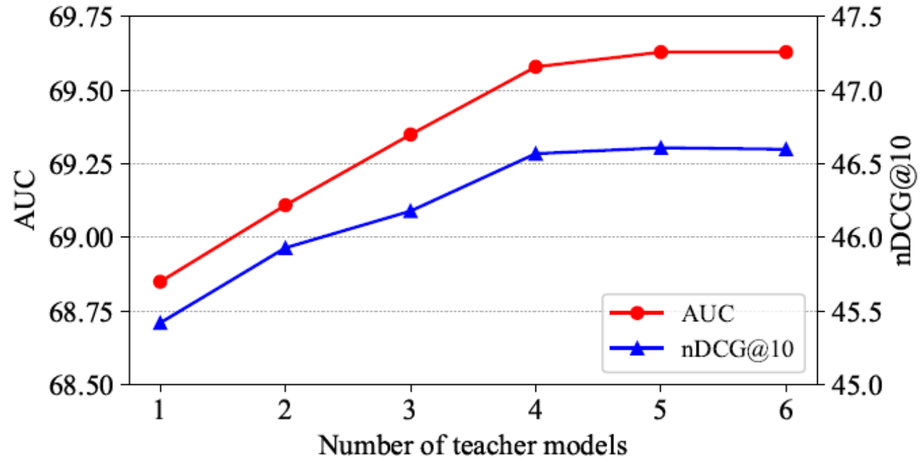
Comparación Adicional

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**

Evaluación Complementaria

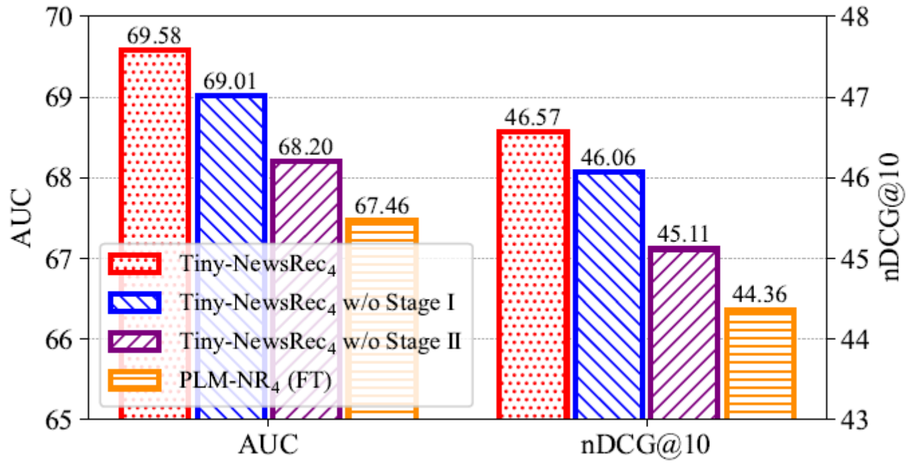
Efectividad del Uso de Modelos de Múltiples Teachers

Se realizan experimentos para analizar cómo influye la cantidad de teachers en el rendimiento de Tiny-NewsRec. Se varía el número de teachers entre 1 y 6 utilizando el modelo student de 4 capas utilizando y el dataset Feeds.



Efectividad de la Destilación de Conocimiento en Dos Etapas

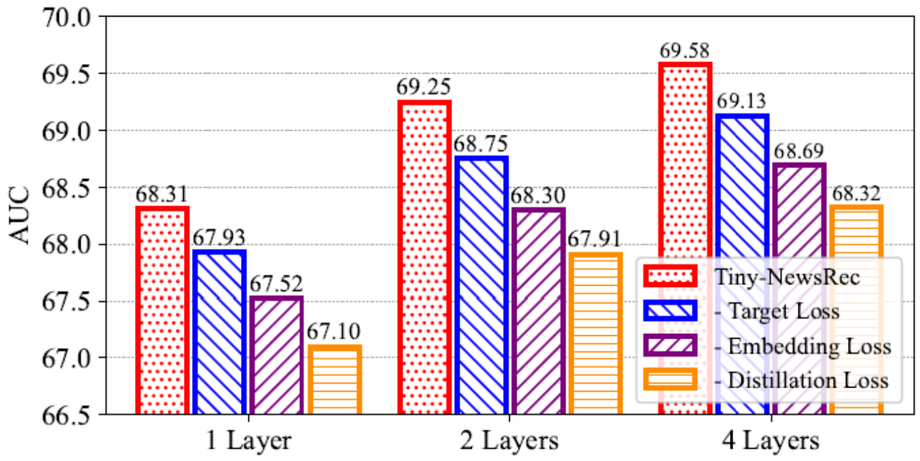
Se verifica la efectividad de cada etapa del método de destilación en dos etapas. Se compara el modelo student de 4 capas de Tiny-NewsRec y sus variantes con una etapa eliminada, utilizando el dataset Feeds.



Evaluación Complementaria

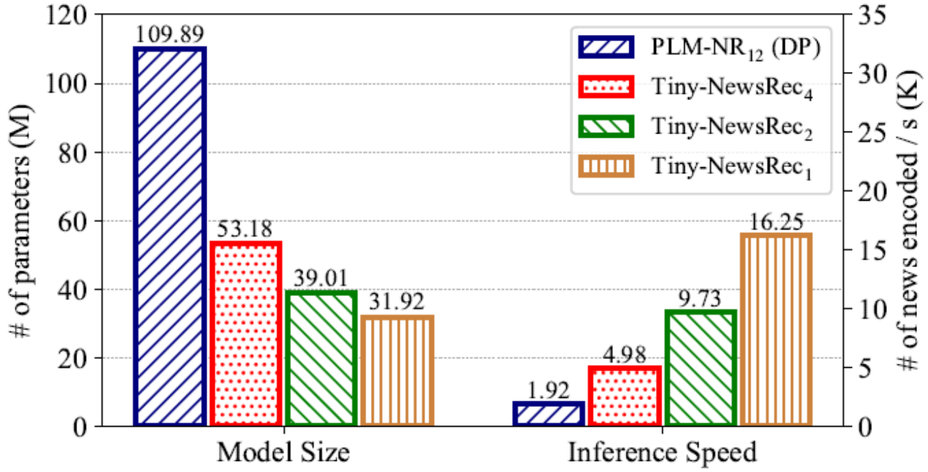
Efectividad de Cada Función de Pérdida

Se explora el impacto de cada parte de la función de pérdida global en el método de destilación, comparando Tiny-NewsRec con sus variantes sin distillation loss, embedding loss y target loss.



Evaluación de la Eficiencia

Dado que la codificación de noticias con un PLM representa el mayor overhead en estos sistemas de recomendación, se mide Velocidad de inferencia y Tamaño del modelo.



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

Yang Yu, Fangzhao Wu, ChuhanWu, Jingwei Yi & Qi Liu

Grupo 10: Felipe Abarca, Alfredo Enrione, Nicolás Estévez

Anexo

MONTAJE EXPERIMENTAL DETALLADO

General Hyper-parameters	
Dimension of query vector in attention network	200
Adam betas	(0.9, 0.999)
Adam eps	1e-8
Domain-specific Post-training	
Negative sampling ratio N	9
Dimension of news title/body representation	256
Batch size	32
Learning rate	1e-6
News Recommendation Finetuning	
Negative sampling ratio S	4
Max number of historical clicked news L	50
Dimension of news/user representation	256
Batch size	32×4
Learning rate	5e-5
Two-stage Knowledge Distillation	
Temperature T_{DP}	1
Temperature T_{FT}	1
Number of teacher models M	4