

CRS 논문 리뷰

전소연 (인턴)

2024년 1월 5일 (금)

Towards Deep Conversational Recommendations

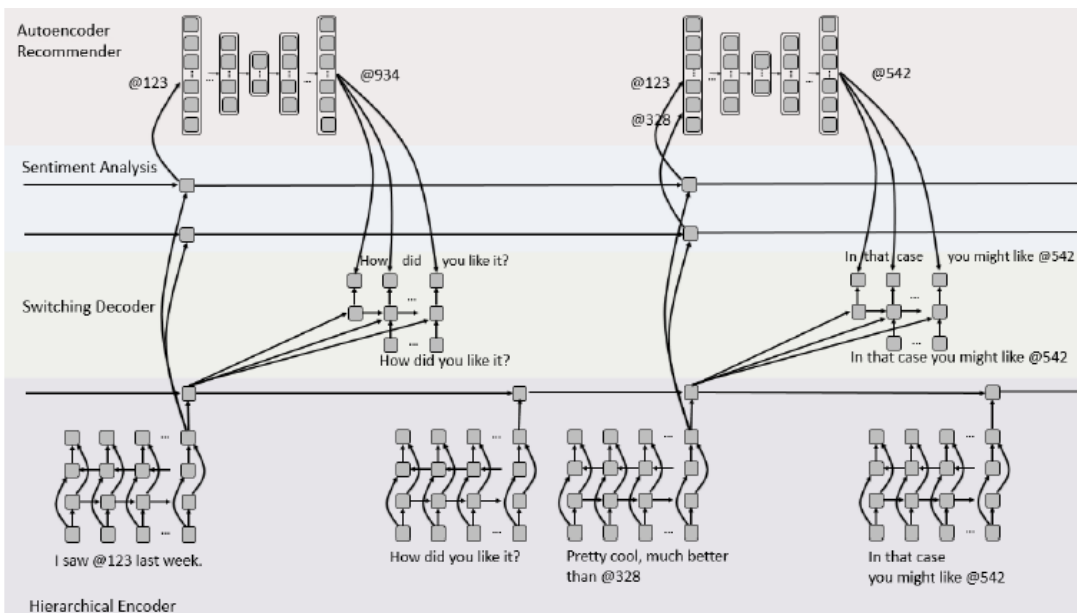
1. Overview

- Recommendations에 대한 real-world dataset을 제공.
- 이 데이터셋을 활용하여 conversational movie recommendations를 위한 neural architecture를 제시.

2. REDIAL(REcommendations through DIALog) Dataset

- Amazon Mechanical Turk(AMT)를 통해 수집한 실제 대화 dataset.
- Seeker와 Recommender의 대화로 구성.
- 하나의 conversation에 utterances 10개 이상, movie mention 4번 이상.
- Movie를 주제로만 대화.
- 각각의 movie에 대해 3개의 label 존재. (movie를 suggest한 사람, seeker가 movie를 봤는지 여부, seeker가 movie를 좋아하는지 여부)

3. Model Architecture



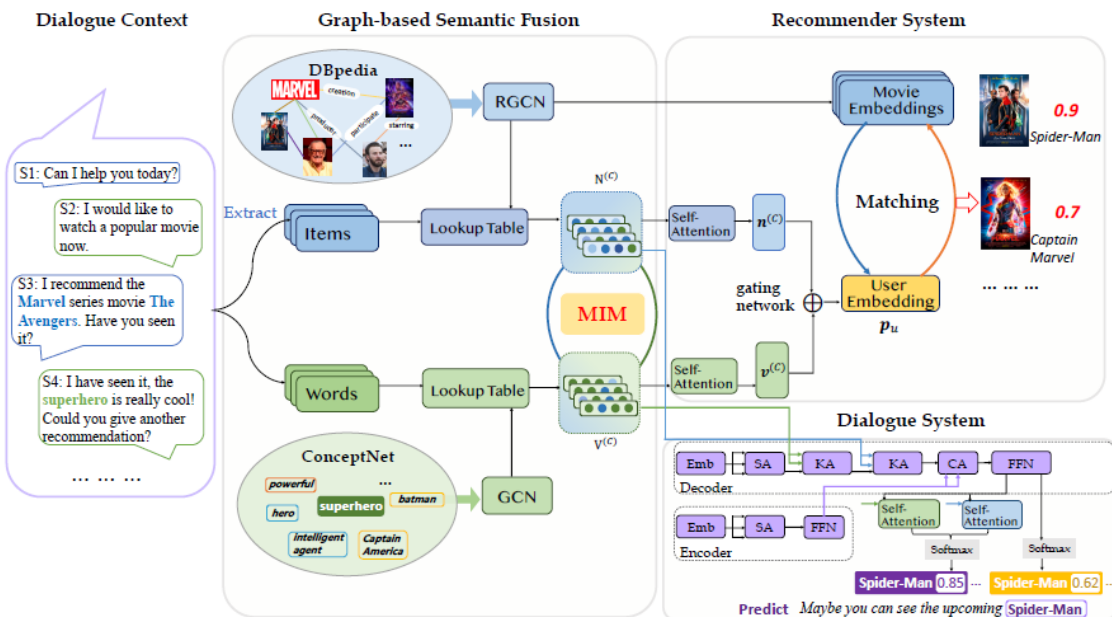
- 1) **Hierarchical Recurrent Encoder** : Utterance를 **Encoding** 하는 역할.
- 2) **RNN for Movie Sentiment Analysis** : Utterance에 언급된 movie에 대하여, seeker와 recommender 각각에 대해 **label**을 예측.
- 3) **Autoencoder Recommender** : movie와 예측한 label을 input으로 받아 **movie**를 추천.
- 4) **Decoder with a Movie Recommendation Switching Mechanism** : 다음 token으로 movie를 생성할지 word를 생성할지 정하는 switching mechanism을 사용하여 더 정확하게 **next token**을 생성.

Improving Conversational Recommender Systems via Knowledge Graph based Semantic Fusion

1. Motivation

- Lack of contextual information -> **Knowledge Graph** 사용.
- word와 item 간의 natural semantic gap 존재 -> **Semantic Fusion**

2. KGSF Structure



Graph-based Semantic Fusion

- item representation : DBpedia에 대해 RGCN을 통해 계산하여 얻음.
- word representation : ConceptNet에 대해 GCN을 통해 계산하여 얻음.
- 이때 MIM(Mutual Information Maximization) loss를 사용해 RGCN과 GCN을 학습시켜 두 representation 간의 차이를 줄이도록 함.

Recommender System

- Word representation과 item representation을 self-attention에 통과시켜 word vector와 item vector를 계산.
- 이 두 벡터에 대해 gate mechanism을 적용해 user embedding을 계산.
- user embedding과 movie embedding 간의 내적을 통해 item에 대한 score 산출.

Dialogue System

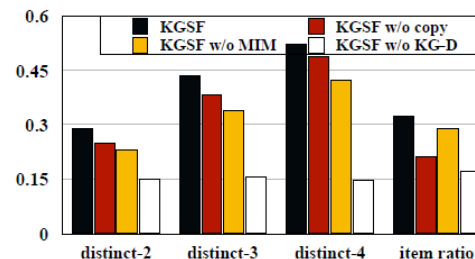
- Transformer를 변형한 KG-enhanced Decoder를 사용
- 이전 layer의 generated words, word-oriented KG, item-oriented KG, context words 순으로 MHA의 key가 되게 함으로써 두 KG에서 얻은 정보를 순차적으로 decoder에 input으로 넣어줌.

3. Experiments – Recommendation task, Conversation task 모두에서 KGSF가 가장 좋은 성능. Ablation Study 통해 모든 component가 성능 향상에 도움이 됨을 알 수 있음.

Test	All data			Cold start		
Models	R@1	R@10	R@50	R@1	R@10	R@50
Popularity	0.012	0.061	0.179	0.020	0.097	0.239
TextCNN	0.013	0.068	0.191	0.011	0.081	0.239
ReDial	0.024	0.140	0.320	0.021	0.075	0.201
KBRD	0.031	0.150	0.336	0.026	0.085	0.242
KGSF	0.039*	0.183*	0.378*	0.039*	0.174*	0.370*
-MIM	0.037	0.175	0.356	0.037	0.158	0.331
-DB	0.027	0.121	0.256	0.030	0.168	0.346

Models	Dist-2	Dist-3	Dist-4	Item Ratio
Transformer	0.148	0.151	0.137	0.194
ReDial	0.225	0.236	0.228	0.158
KBRD	0.263	0.368	0.423	0.296
KGSF	0.289*	0.434*	0.519*	0.325*

Models	Fluency	Informativeness
Transformer	0.92	1.08
ReDial	1.37	0.97
KBRD	1.18	1.18
KGSF	1.54*	1.40*

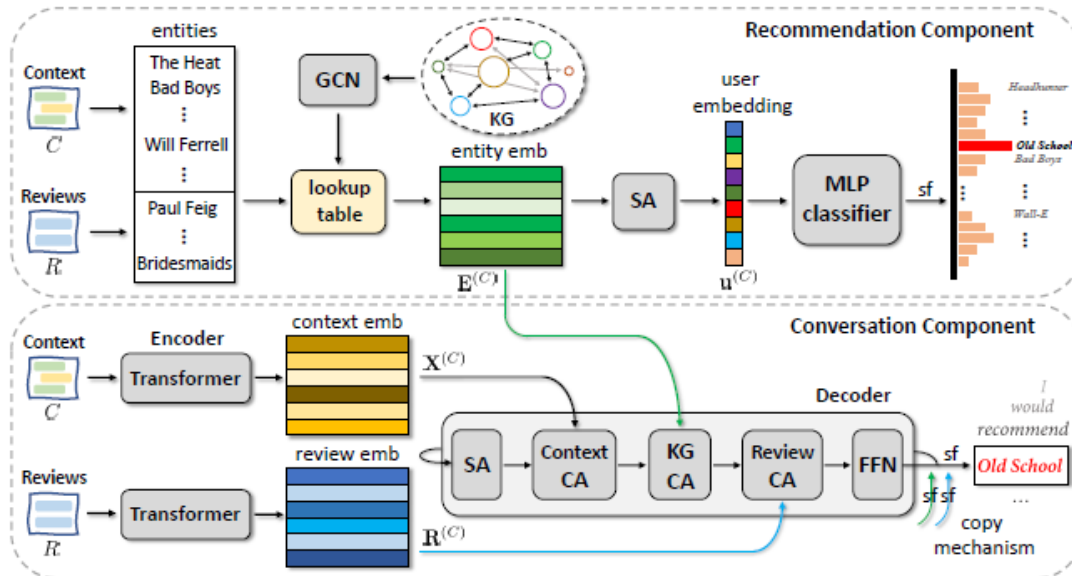


RevCore: Review-augmented Conversational Recommendation

1. Motivation

- dialogue history가 짧거나 unfamiliar한 item일 경우 item information 부족 → External knowledge 중 **review**를 활용하자.

2. RevCore Structure



Sentiment-aware-Retrieval

- context와 sentiment가 일치하는 review를 review database에서 retrieve
- review database 안의 review의 sentiment는 transformer 기반의 sentiment predictor를 사용하여 예측.

Recommendation Component (Review-augmented)

- KG(Dbpedia)에서 GCN 통해 candidate entity embedding dictionary 구축.
- C와 R에 존재하는 entity에 대해 위의 dictionary에서 embedding을 얻음.
- entity embedding을 self-attention에 통과시켜 user embedding을 얻음.
- 이 user embedding을 MLP와 softmax를 통과시켜 movie들에 대한 확률 분포를 얻음.

Conversation Component (Review-augmented)

- Context와 Review 각각을 transformer에 통과시켜 embedding 얻음.
- 이전 time의 decoder output, context embedding, entity embedding, review embedding 순으로 MHA의 key로 사용.
- 마지막으로 softmax를 통과시켜 줌으로써 token에 대한 확률분포를 얻고 copy mechanism을 적용하여 response를 생성.

3. Experiments

Models	R@1	R@10	R@50
Redial	2.4	14.0	32.0
KBRD	3.1	15.0	33.6
KGSF	3.9	18.3	37.8
RevCore (-KG)	4.2	22.7	43.3
RevCore (+KG)	6.1	23.6	45.4

Models	Dist-2	Dist-3	Dist-4	PPL
Trans	0.148	0.151	0.137	17.0
Redial	0.225	0.236	0.228	28.1
KBRD	0.263	0.368	0.423	17.9
KGSF	0.289	0.434	0.519	9.8
RevCore (-KG)	0.373	0.527	0.615	10.7
RevCore (+KG)	0.424	0.558	0.612	10.2

Models	Coherence	Fluency	Informat
Trans	0.189	0.226	0.115
Redial	0.225	0.455	0.228
KBRD	0.263	0.468	0.283
KGSF	0.324	0.502	0.332
RevCore (-KG)	0.556	0.493	0.682
RevCore (+KG)	0.601	0.567	0.718

Models	Dist-2	Dist-3	Dist-4	PPL
RevCore (+KG)	0.424	0.558	0.612	10.2
-revCP	0.353	0.443	0.503	10.0
-revRA	0.328	0.428	0.516	13.2
-revEN	0.394	0.534	0.586	10.8

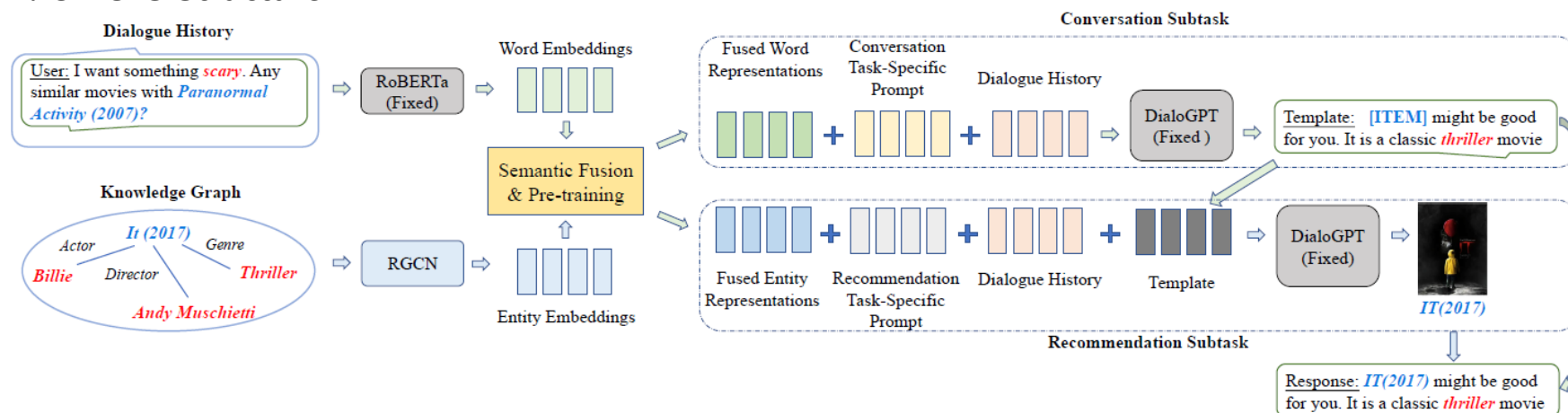
Towards Unified Conversational Recommender Systems via Knowledge-Enhanced Prompt Learning

1. Motivation

Root of Semantic inconsistency 존재.

- Recommender module과 Dialogue module의 구조가 다르기 때문 → **Prompt Learning**을 통해 **Unified manner**로 두 module을 훈련시키자.
- 한 module의 구조를 다른 한 모듈이 사용할 수 없기 때문 → **Template**을 사용하자.

2. UniCRS Structure



- Dialogue History와 KG 각각에 대해 embedding을 얻고 semantic fusion과 pre-training을 수행.
- Pre-trained Prompt Tuning 아이디어를 그대로 사용. recommendation subtask와 conversation subtask를 PLM의 downstream task로 세팅하고 각각의 prompt를 pre-training
- 위와 같이 각각의 subtask에 대해 prompt를 구성해 DialoGPT에 주어 각각 template과 movie 추천을 얻음.
- template에 Recommendation subtask에서 얻은 recommendation을 포함시켜 response

3. Experiments – Recommendation task, Conversation task 모두에서 UniCRS가 가장 좋은 성능. Ablation Study 통해 모든 component가 성능 향상에 도움이 됨을 알 수 있음.

Datasets	ReDial			INSPIRED		
Models	R@1	R@10	R@50	R@1	R@10	R@50
ReDial	0.023	0.129	0.287	0.003	0.117	0.285
KBRD	0.033	0.175	0.343	0.058	0.146	0.207
KGSF	0.035	0.177	0.362	0.058	0.165	0.256
GPT-2	0.023	0.147	0.327	0.034	0.112	0.278
DialoGPT	0.030	0.173	0.361	0.024	0.125	0.247
BERT	0.030	0.156	0.357	0.044	0.179	0.328
BART	0.034	0.174	0.377	0.037	0.132	0.247
UniCRS	0.051*	0.224*	0.428*	0.094*	0.250*	0.410*

Datasets	ReDial			INSPIRED		
Models	Dist-2	Dist-3	Dist-4	Dist-2	Dist-3	Dist-4
ReDial	0.225	0.236	0.228	0.406	1.226	2.205
KBRD	0.281	0.379	0.439	0.567	2.017	3.621
KGSF	0.302	0.433	0.521	0.608	2.519	4.929
GPT-2	0.354	0.486	0.441	2.347	3.691	4.568
DialoGPT	0.476	0.559	0.486	2.408	3.720	4.560
BART	0.376	0.490	0.435	2.381	2.964	3.041
UniCRS	0.492*	0.648*	0.832*	3.039*	4.657*	5.635*

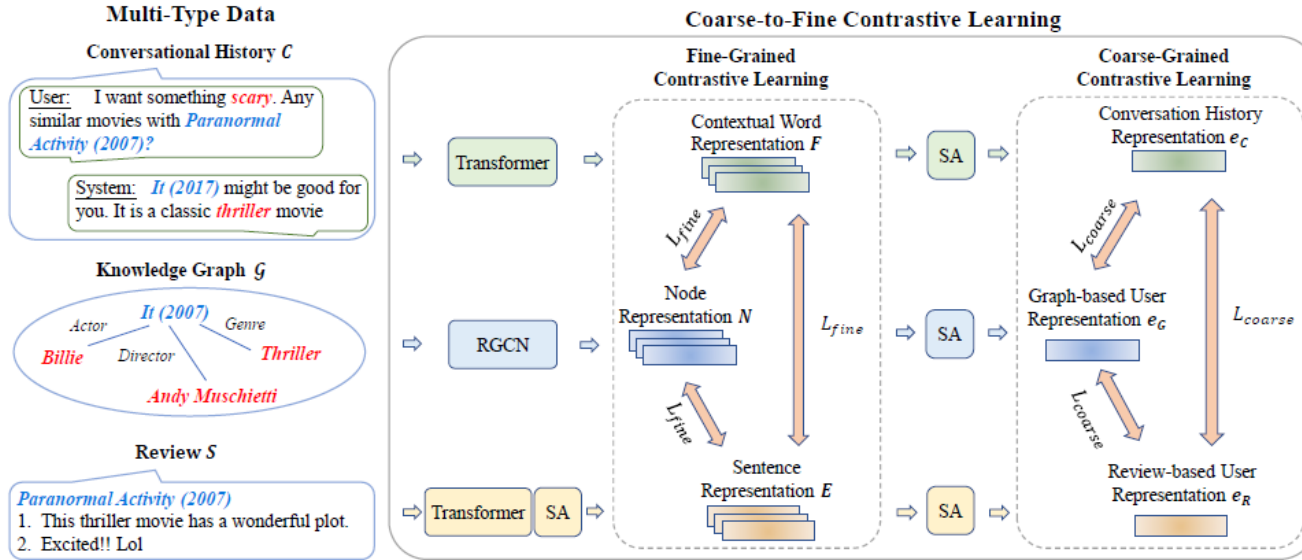
Models	Fluency	Informativeness
ReDial	1.31	0.98
KBRD	1.21	1.16
KGSF	1.49	1.39
GPT-2	1.62	1.48
DialoGPT	1.68	1.56
BART	1.63	1.43
UniCRS	1.72*	1.64*

C2-CRS: Coarse-to-Fine Contrastive Learning for Conversational Recommender System

1. Motivation

- Natural semantic gap -> **Bridging semantic gap by general method** (Dialogue history, DBpedia KG, Review 세 가지 데이터에 대해 semantic gap을 줄였음.)
- Multi-grained form of context data -> **Multi-grained semantic alignment**

2. Method



Encoding Multi-type Data

- 각각의 데이터에 대해 두 종류의 representation을 생성.

Coarse-to-Fine Contrastive Learning (Pre-training)

- Encoding된 Multi-type data의 representation space를 효과적으로 fusion하기 위함. → multi-type data의 multi-grained correlations에 기반.
- Fine-Grained Contrastive Learning : Fine-grained characteristics에 대한 specific user preference를 반영.
- Coarse-grained Contrastive Learning : overall user preference를 반영.

Fine-tuning Recommendation Module

- User Representation과 Item Representation들을 내적하여 모든 item에 대해 ranking score를 계산.

Fine-tuning Response Generation Module

- KGSF의 transformer decoder 구조를 변형.
- Copy network를 사용하여 enhance informative response generation

3. Experiments – Recommendation task, Conversation task 모두에서 C2-CRS가 가장 좋은 성능. Ablation Study 통해 모든 component가 성능 향상에 도움이 됨을 알 수 있음.

Dataset	ReDial			TG-ReDial		
Models	R@1	R@10	R@50	R@1	R@10	R@50
Popularity	0.011	0.054	0.183	0.0004	0.003	0.014
TextCNN	0.013	0.068	0.191	0.003	0.010	0.024
ReDial	0.024	0.140	0.320	0.000	0.002	0.013
KBRD	0.031	0.150	0.336	0.005	0.032	0.077
KGSF	0.039	0.183	0.378	0.005	0.030	0.074
KECRS	0.021	0.143	0.340	0.002	0.026	0.069
RevCore	0.046	0.220	0.396	0.004	0.029	0.075
C ² -CRS	0.053*	0.233*	0.407*	0.007*	0.032*	0.078*

Dataset	ReDial			TG-ReDial		
Models	Dist-2	Dist-3	Dist-4	Dist-2	Dist-3	Dist-4
Transf	0.067	0.139	0.227	0.053	0.121	0.204
ReDial	0.082	0.143	0.245	0.055	0.123	0.215
KBRD	0.086	0.153	0.265	0.045	0.096	0.233
KGSF	0.114	0.204	0.282	0.086	0.186	0.297
KECRS	0.040	0.090	0.149	0.047	0.114	0.193
RevCore	0.092	0.163	0.221	0.043	0.105	0.175
C ² -CRS	0.163*	0.291*	0.417*	0.189*	0.334*	0.424*

Models	Fluency	Informativeness
Transf	0.97	0.92
ReDial	1.35	1.04
KBRD	1.23	1.15
KGSF	1.48	1.37
KECRS	1.39	1.19
RevCore	1.52	1.34
C ² -CRS	1.55*	1.47*

Large Language Models as Zero-Shot Conversational Recommenders

- **Overview**

- 학습 없이 zero-shot으로 LLM을 CR에 적용.
- LLM의 recommendation ability를 평가.

- **Data**

- 더 정확한 recommendation ability 평가를 위해 새로운 dataset 제안.
- 다양한 domain, 언어, 문화를 포함하게끔.
- Reddit으로 real user dialogue를 수집. 기존 CRS dataset의 50배 크기.

- **Evaluation**

- 기존 CRS evaluation에서는 이미 언급된 repeated item과 new item에 대해 동일하게 평가했기 때문에 recommendation이 repeated item에 편향될 수 있었음. → Item set을 두 가지 category (Repeated Items / New Items)로 나누어 각각 성능 비교하자 기존 평가 방식에서는 CRS model 성능 우수, but new item에 대해서는 성능이 하락.
- zero-shot LLM이 기존의 fine-tuned CRS model을 outperform

- **Analysis**

- Conversational Recommender로서 zero-shot LLM의 우수성을 보이기 위해 model 측면과 data 측면에서 다양한 분석을 수행하였음.

TALLRec: An Effective and Efficient Tuning Framework to Align Large Language Model with Recommendation

- **Abstract**

- LLM 훈련과 recommendation task 사이의 gap → Recommendation data로 LLM을 tuning함으로써 Large Recommendation Language Model을 만들자.

- **Conclusion**

- 제안한 TALLRec 프레임워크를 사용해 train한 LLM의 경우 기존 모델을 outperform하고 cross-domain generalization ability를 가짐.

Reference

https://www.youtube.com/watch?v=qKB_esBvOig
https://www.youtube.com/watch?v=aOWGFnnT_X8
<https://www.youtube.com/watch?v=1YZVNVttXaY>