Graph Mining in Recommender Systems

- - A Tutorial at WISE 2021

- Speakers:
- Dr Hongxu Chen
- Miss Yicong Li
- Mr Haoran Yang

University of Technology of Sydney (UTS)

UTS

DSMIL
Data Science & Machine Intelligence Lab

Agenda

- Graph Representation Learning and its applications in Recommendation Systems

  by Hongxu Chen, 14:15 – 14:45 October 27, 2021 (Wednesday)

- Graph-based Explainable Recommender Systems

  by Yicong Li, 14:45 – 15:15 October 27, 2021 (Wednesday)

- Graph Contrastive Learning for Recommender Systems

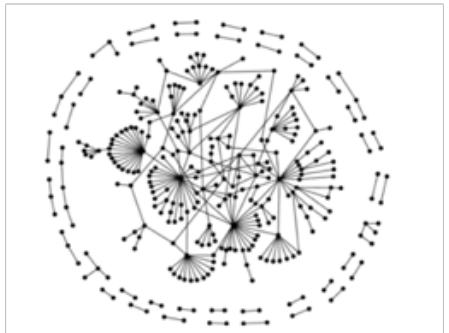
  by Haoran Yang, 15:15 – 15:45 October 27, 2021 (Wednesday)

1. Graph Representation Learning and its applications in Recommendation Systems

Networks are ubiquitous

Social networks

Road networks



Communication graphs

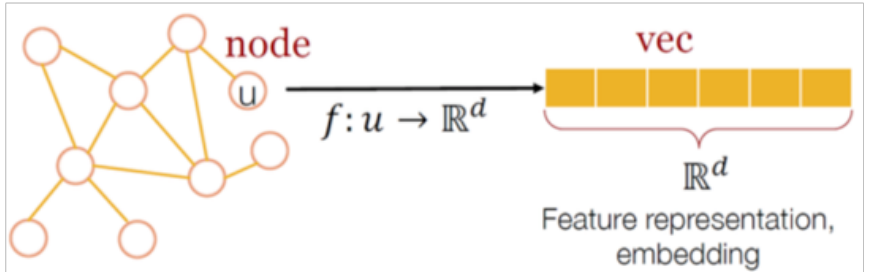
Information networks:
Web & citations

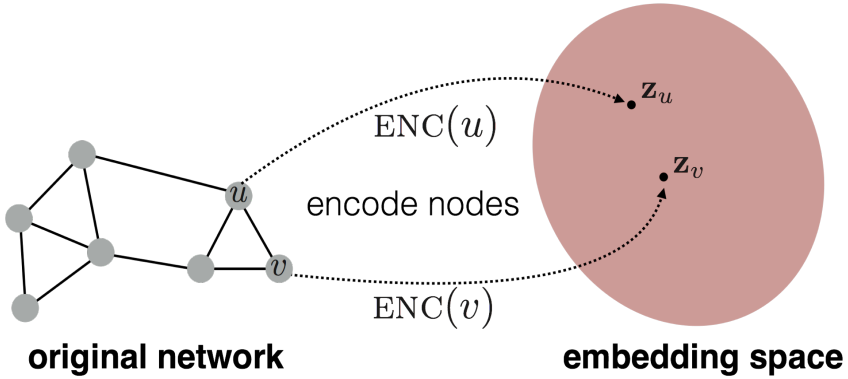
Internet

Networks of neurons

Representing networks by vectors

- Graph Representation Learning.
  - Also known as Graph Embedding or Network Embedding.
  - Low-dimensional vector for vertices.
  - Effectively preserve network structure.





  - Downstream data mining tasks on graphs:
    - ✓ Link prediction.
    - ✓ Node classification.
    - ✓ Recommendation.
    - ✓ …

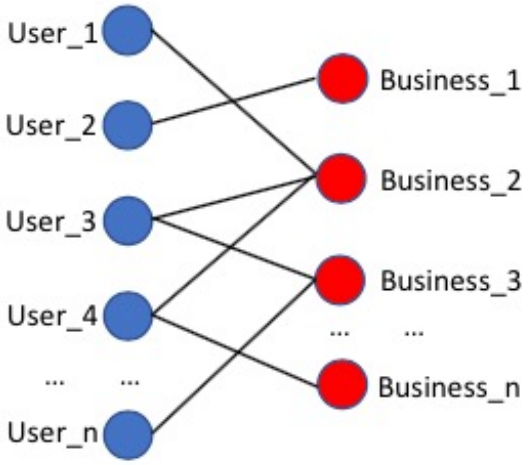
Graph Embedding Methods

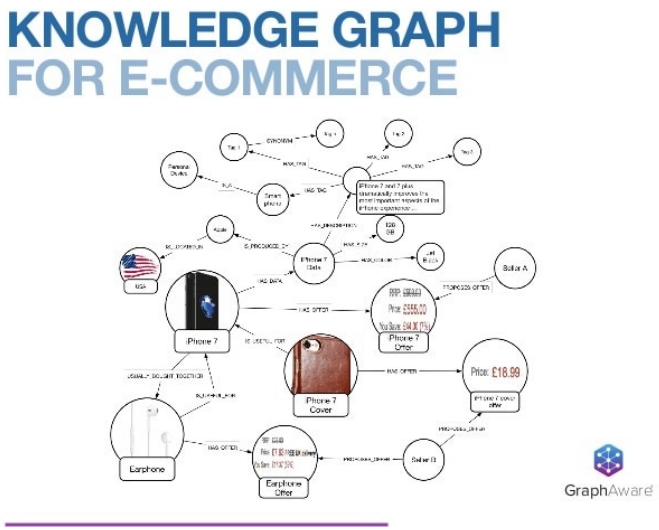
- Early Methods (Matrix Factorization-based)
  - E.g., MDS, IsoMap, Spectral Clustering, Laplacian Eigenmap, etc.
- Random-walk based Embedding Methods
  - E.g., DeepWalk, Node2Vec, Metapath2Vec, etc.
- Graph Neural Networks
  - E.g., GCNs, SAGE, GAT, Graph Autoencoder, etc.

Handy Materials:

[1]. http://web.stanford.edu/class/cs224w/

Recommendation Systems as Graphs



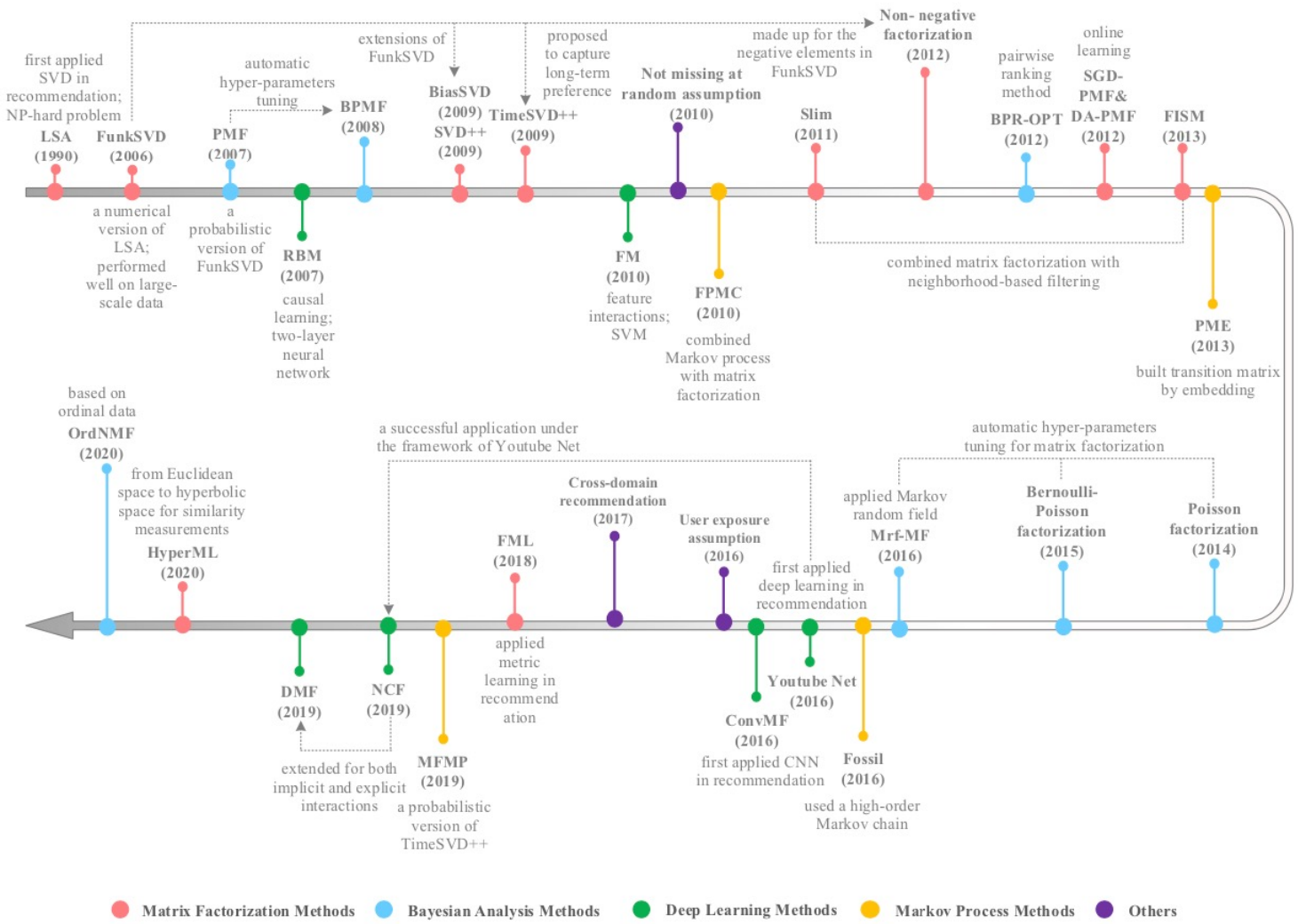


https://images.app.goo.gl/YjzjovQnvZgwwdx56

As Bipartite Graph Mining

- Matrix Factorization
- Factorization Machines
- Bayesian Analysis (PMF, BPR)
- Deep Learning (NCF, DMF, DeepFM)

Timeline of development of Bipartite Modelling Approaches



Deng, Y. (2021). Recommender systems based on graph embedding techniques: A comprehensive review. *arXiv preprint arXiv:2109.09587*.

As General Graph Modelling

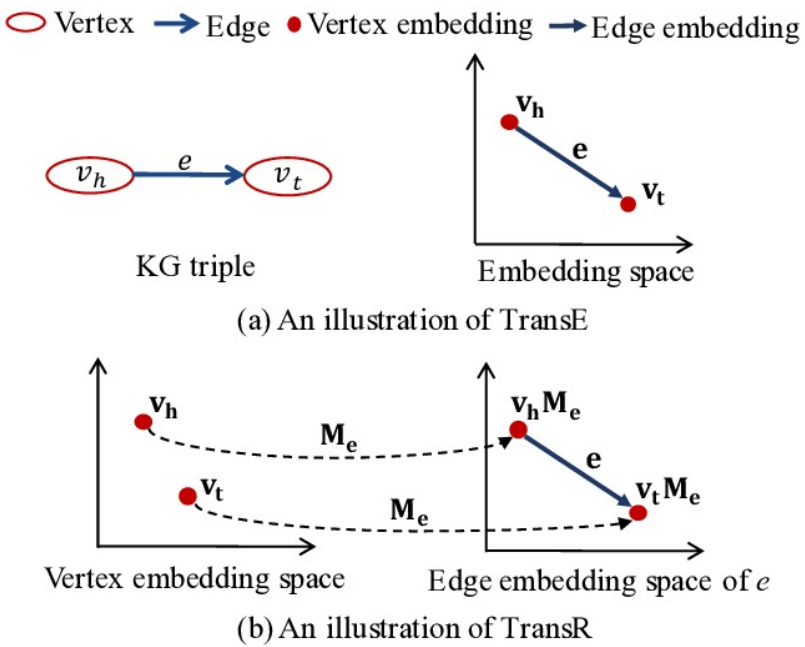
- Trans Family (e.g., TransE, TransR, TransH, TransG, etc)
- Random Walk and Meta-path
- Deep Learning

Trans Family

- Model triplets (*h,r,t*) as basic elements in a graph.

For example:

- TransE models a relation *r* as a translation from head entity *h* to tail entity *t.*
- TransH maps each pair of (*h, t*) to multiple relation-specific hyper-planes distinguishing the diverse relations between nodes.
- TransR maps nodes and relations separately into a node space and multiple different relation spaces corresponding to the diverse relations between head-tail pairs, respectively.

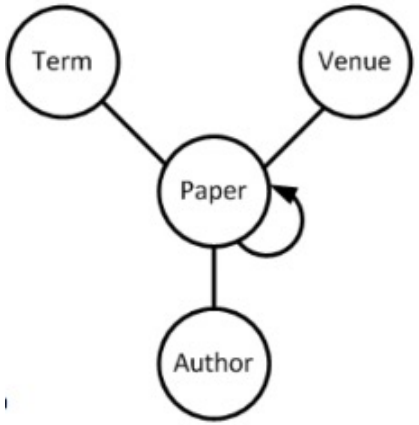


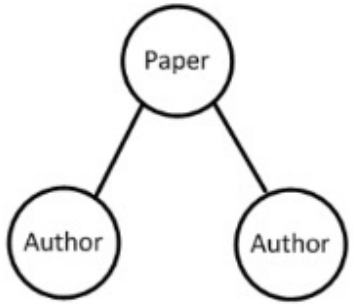
Random Walk and Meta-path

- Meta-path

➢Meta-level description of a path between two objects (Sun, et al.).

➢Different Meta-Paths tell different Semantics.





Random Walk and Meta-path

- Random Walk as paths.
- Encode the paths as sentences (Inspired by advances in NLP).

➢Word2Vec: Skip-gram and Continuous Bag of Words (CBOW)
➢Negative Sampling

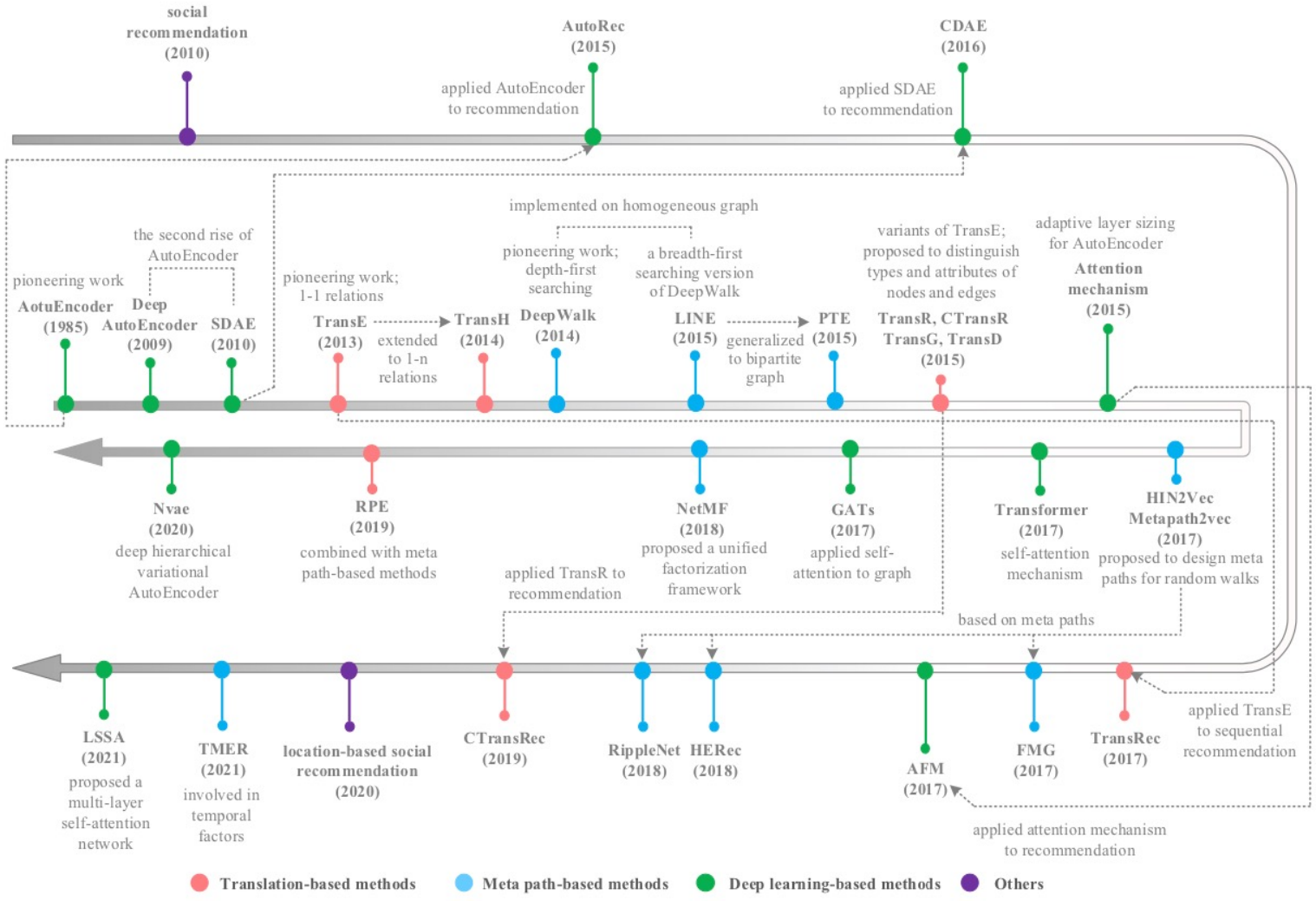
- Graphs Embedding techniques recap:

➢LINE
➢Randwalk
➢Deepwalk
➢Node2Vec
➢Metapath2Vec
➢BINE (can also classified as Bipartite Network Embedding based method)

Deep Learning based approaches

Autoencoders	SDAE	Transformer
GCNs	SAGE	GAT

Timeline of General Graph Embedding based Approach



Deng, Y. (2021). Recommender systems based on graph embedding techniques: A comprehensive review. *arXiv preprint arXiv:2109.09587*.

Where are we in embedding spaces? A Comprehensive Analysis on Network Embedding Approaches for Recommender Systems

- Latent Space Models
  - Most are built in Euclidean space.
    - Suffer from high distortion.
    - Need to increase the latent size in order to reduce the distortion.
    - Resources needed to train and store the model also increases.

- Hyperbolic Space:
  - A promising new latent space to solve the above issues.
    - Expands faster than Euclidean space.
    - Preserve the hierarchies of data.

Where are we in embedding spaces? A Comprehensive Analysis on Network Embedding Approaches for Recommender Systems

❑ Issues for Existing Literatures on Hyperbolic Space:

- ❖ Practitioners are indiscriminately attempting to transfer variants of existing algorithms into hyperbolic geometry in regardless of having strong motivations.
- ❖ Analysis and guidance about when to use hyperbolic space are missing.

❑ Our Contributions:

- ❖ We provide theoretical analysis and empirical results to validate our three hypotheses about the effectiveness of hyperbolic space.
- ❖ We address the drawbacks of hyperbolic space, and give comments on when and where to use hyperbolic space.
- ❖ We propose a metric learning based social recommendation method SCML and its hyperbolic version HSCML and validate our hypotheses on them. We also show that hyperbolic space has state-of-the-art performance by comparing HSCML with other baselines on two benchmark datasets.

Where are we in embedding spaces? A Comprehensive Analysis on Network Embedding Approaches for Recommender Systems

❑ Hypotheses

- ❖ Distance models are more suited for learning hyperbolic embeddings than projection models.
  - ➢ The distribution of embeddings is different after convergence.
  - ➢ Embeddings in projection models gather around the origin, making little use of the whole space.
  - ➢ Distance models can learn hierarchical information easier.

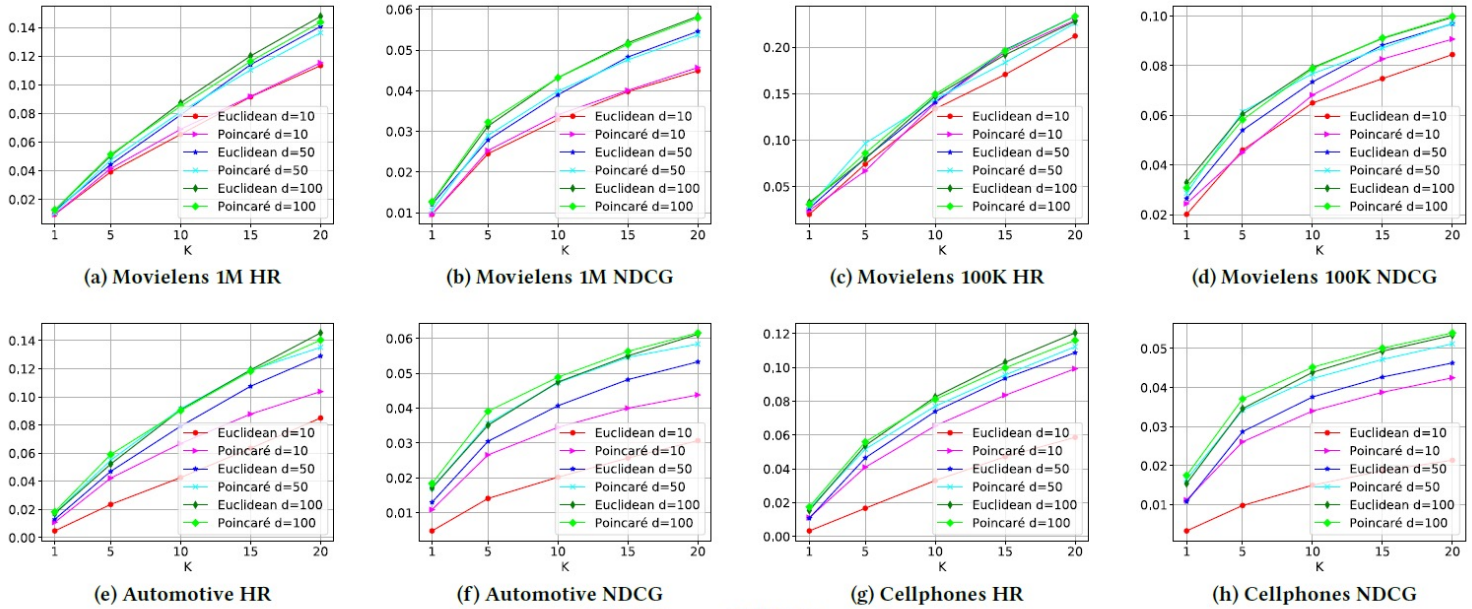
- ❖ Hyperbolic space is more powerful compared to Euclidean space when the density of dataset is small.
  - ➢ Hyperbolic space can boost learning using the extracted hierarchical information.

- ❖ The performance of hyperbolic space is better than Euclidean space when the latent size is small, but as the latent size increases, the performance of Euclidean space becomes comparable with hyperbolic space.
  - ➢ The latent size needed to embed certain amount of information is smaller for hyperbolic space.
  - ➢ With large enough latent size, Euclidean space will also have a very small distortion.

Where are we in embedding spaces? A Comprehensive Analysis on Network Embedding Approaches for Recommender Systems

❑ Experiments (General Item Recommendation)

❖ High Density Datasets vs. Low Density Datasets



(a) Movielens 1M HR (b) Movielens 1M NDCG (c) Movielens 100K HR (d) Movielens 100K NDCG

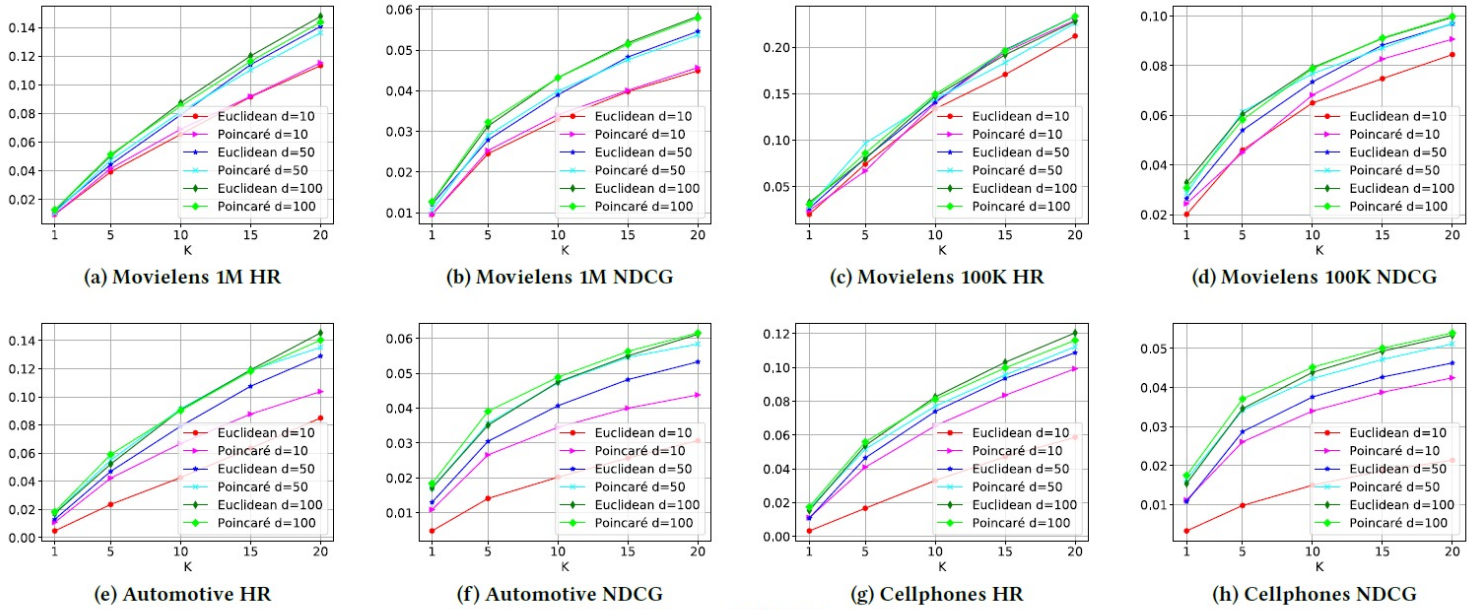
(e) Automotive HR (f) Automotive NDCG (g) Cellphones HR (h) Cellphones NDCG

Figure 2: CML HRs and NDCGs on four datasets.

Where are we in embedding spaces? A Comprehensive Analysis on Network Embedding Approaches for Recommender Systems

❑ Experiments (General Item Recommendation)

❖ Influence of the Latent Size



(a) Movielens 1M HR (b) Movielens 1M NDCG (c) Movielens 100K HR (d) Movielens 100K NDCG

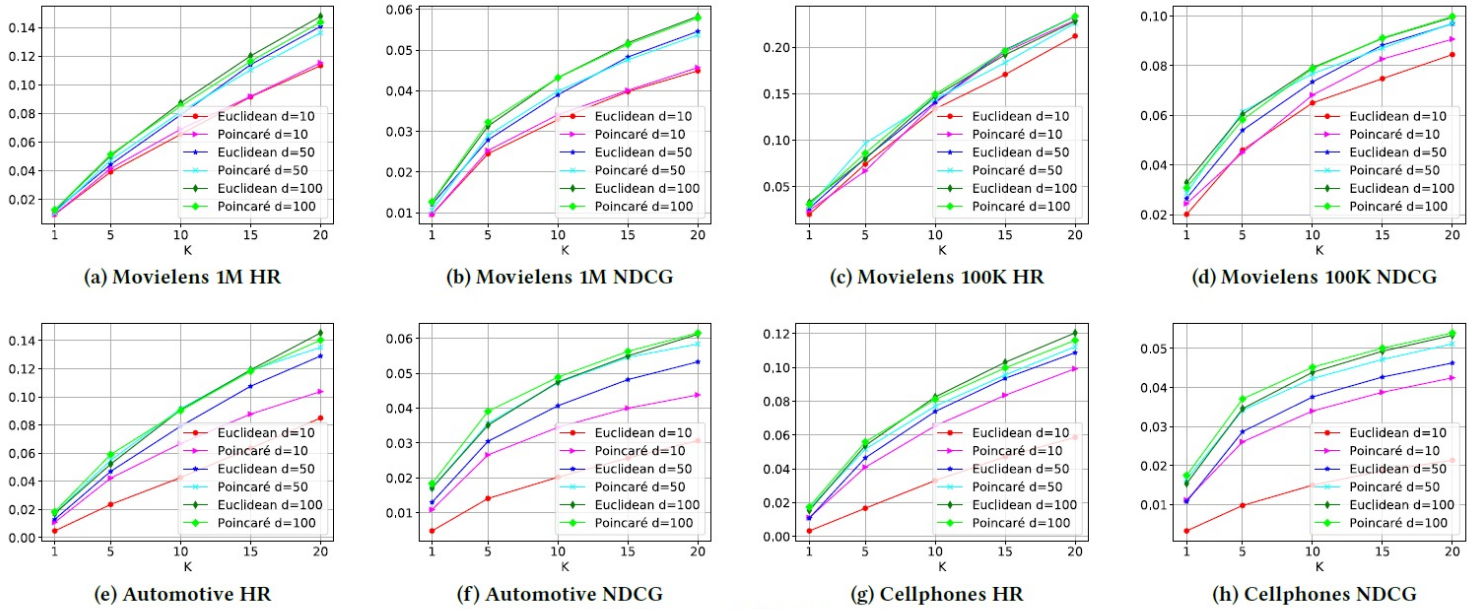
(e) Automotive HR (f) Automotive NDCG (g) Cellphones HR (h) Cellphones NDCG

Figure 2: CML HRs and NDCGs on four datasets.

Where are we in embedding spaces? A Comprehensive Analysis on Network Embedding Approaches for Recommender Systems

❑ Experiments (General Item Recommendation)

❖ Distance Model vs. Projection Model



(a) Movielens 1M HR (b) Movielens 1M NDCG (c) Movielens 100K HR (d) Movielens 100K NDCG

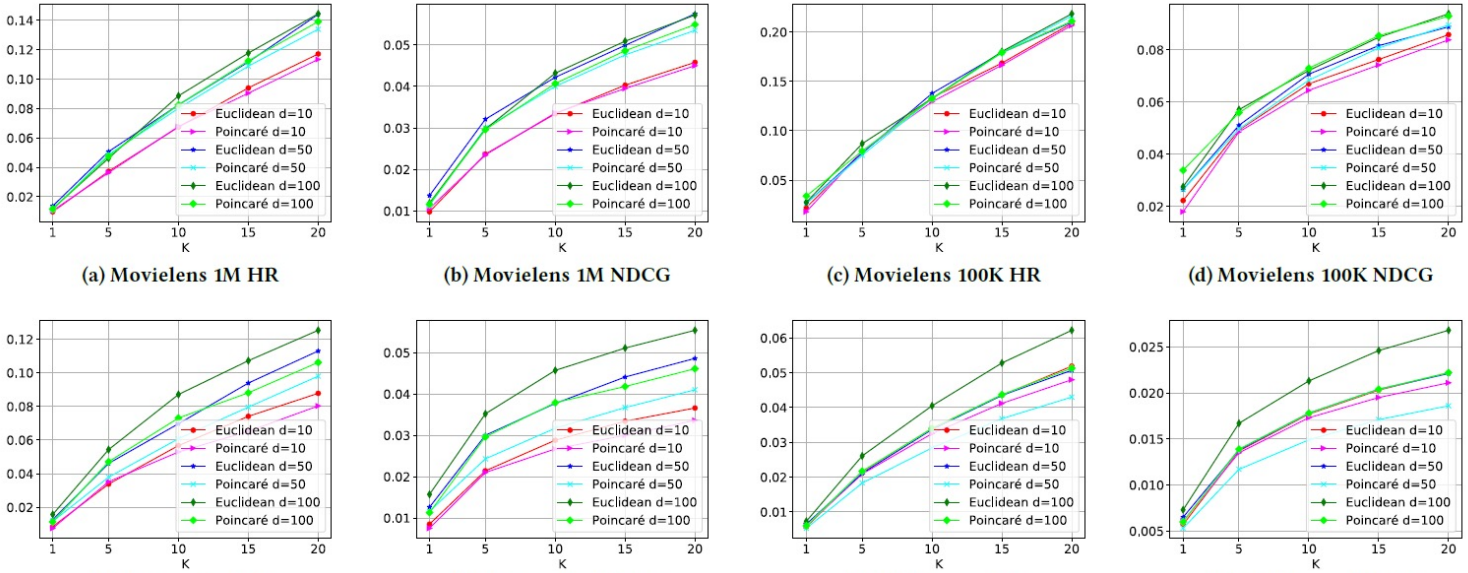
(e) Automotive HR (f) Automotive NDCG (g) Cellphones HR (h) Cellphones NDCG

Figure 2: CML HRs and NDCGs on four datasets.

Where are we in embedding spaces? A Comprehensive Analysis on Network Embedding Approaches for Recommender Systems

❑ Experiments (General Item Recommendation)

❖ Distance Model vs. Projection Model



(a) Movielens 1M HR (b) Movielens 1M NDCG (c) Movielens 100K HR (d) Movielens 100K NDCG

(e) Automotive HR (f) Automotive NDCG (g) Cellphones HR (h) Cellphones NDCG

Figure 3: MF-BPR HRs and NDCGs on four datasets.

Where are we in embedding spaces? A Comprehensive Analysis on Network Embedding Approaches for Recommender Systems

❑ Drawbacks of Hyperbolic Space

- ❖ High computational complexity.
- ❖ Invalid values.

❑ Comments on Using Hyperbolic Space

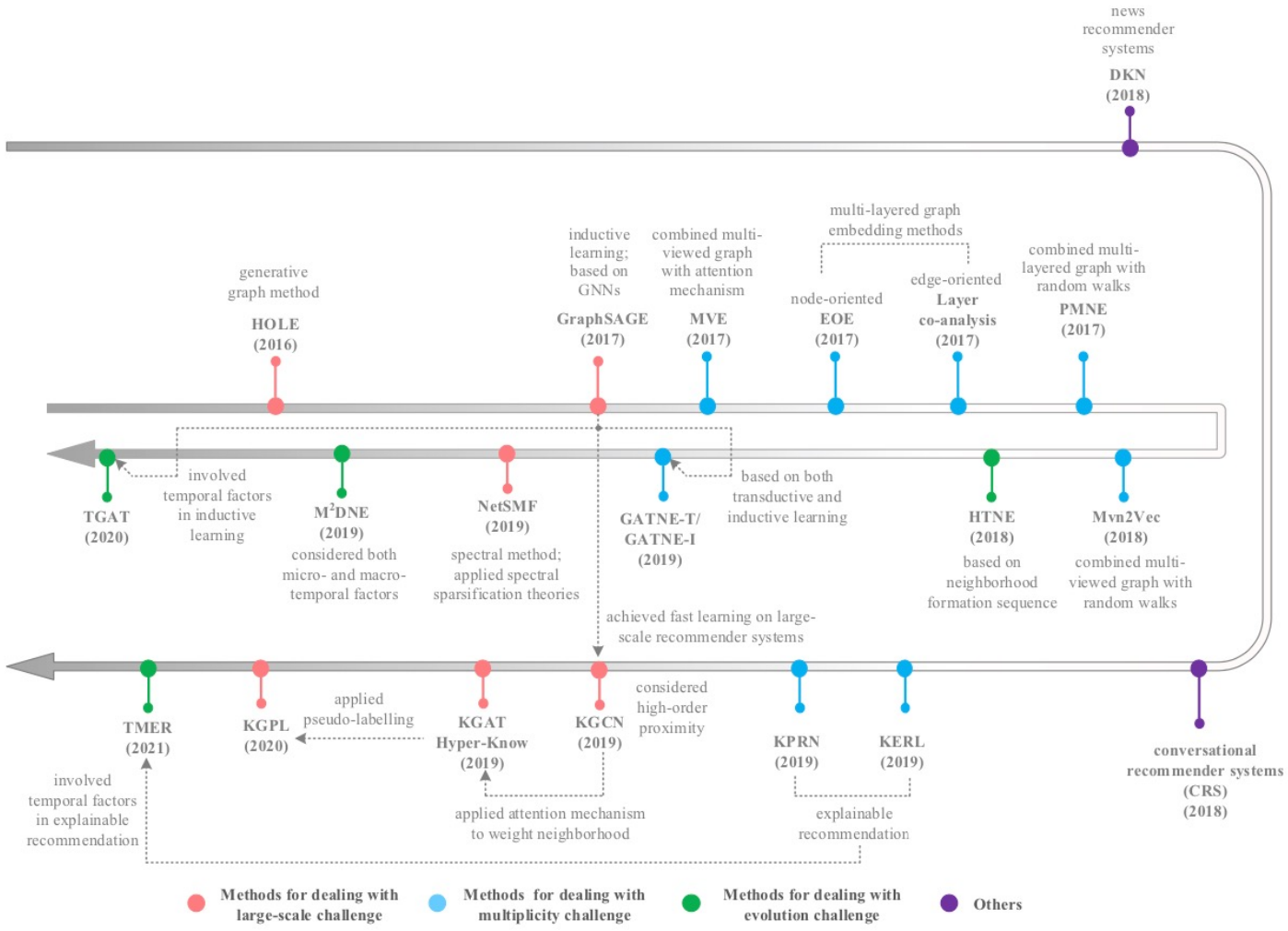
- ❖ Distance models are more suited for learning hyperbolic embeddings than projection models.
- ❖ If the density of the dataset is large, Euclidean space should be a better choice because it has a comparable performance with hyperbolic space and the computational complexity is low; when the density is low, hyperbolic space is more preferable because it will have a much better performance than Euclidean space.
- ❖ Choose an appropriate latent size. In most cases, hyperbolic embeddings only need a relatively small latent size to achieve good performance, which can help to save resources.

Others and Challenges

Three ground challenges:

- Heterogeneity
- Scalability
- Dynamics

Timeline of key developments regrading the challenges

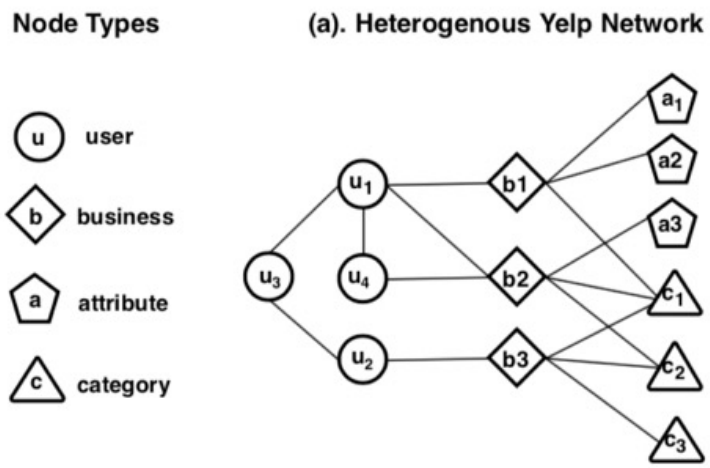


Deng, Y. (2021). Recommender systems based on graph embedding techniques: A comprehensive review. *arXiv preprint arXiv:2109.09587*.

Heterogeneity

Homogenous Networks

- ❖ Single-Typed Nodes.
- ❖ Single-Typed Links.

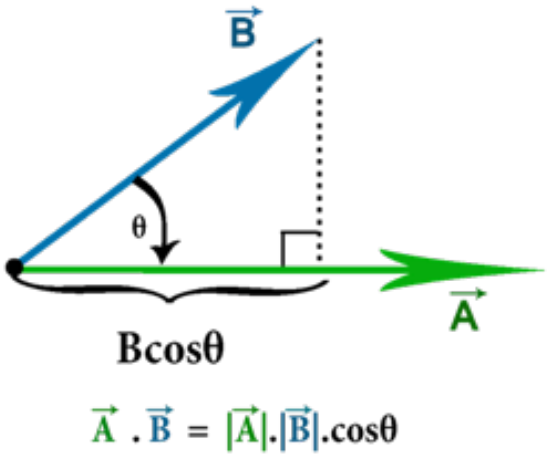


The Real World: Heterogenous Networks

- ❖ Multiple object types.
- ❖ Multiple link types.

E.g., vertex u1 is close to both b2 and u3, but these relationships have different semantics. b2 is a business visited by user u1, while u3 is a friend of u1.

Projected Metric Embedding (KDD18)



- Previous attempts.
  To model semantic-specific relationships:
  - Metapath2vec (Dong et. al., KDD 2017)
  - EOE (Xu et. al., WSDM 2017)
  - Dot product is used to compute the proximity between different types of nodes
  - Not a metric based distance
  - Violates the crucial triangle inequality

- Node A is close to Node C
- Node B is close to Node C

Node A is also close to Node B

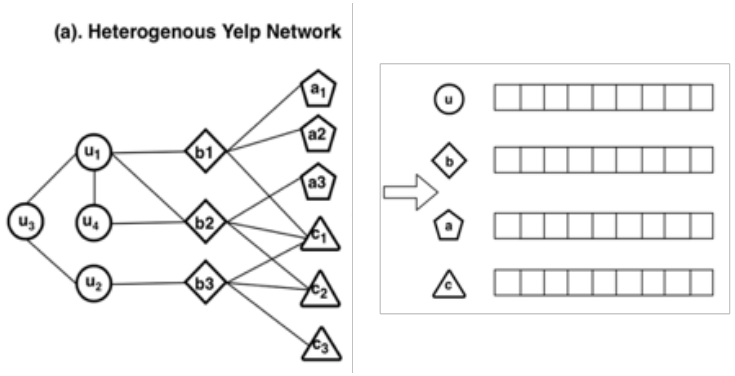
Therefore, existing HIN embedding methods (e.g., Metapath2vec and EOE)

- Can only capture local structures (both A and B are close to C)
- But fail to capture the second-order proximities. (A and B are also close)

Projected Metric Embedding (KDD18)

Our proposed method

- ❑ Projected Metric Embedding for HIN
  - ❖ Simultaneously preserves:
    - ➢ The first-order proximities between nodes
    - ➢ And the second-order proximities between nodes



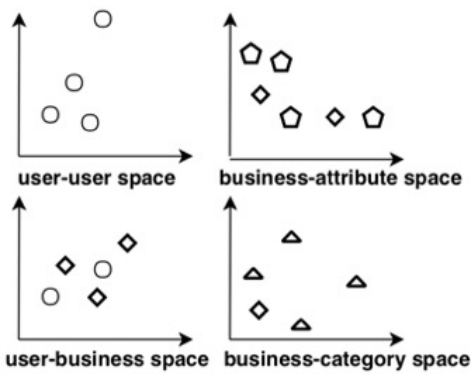
Such that, in the latent space

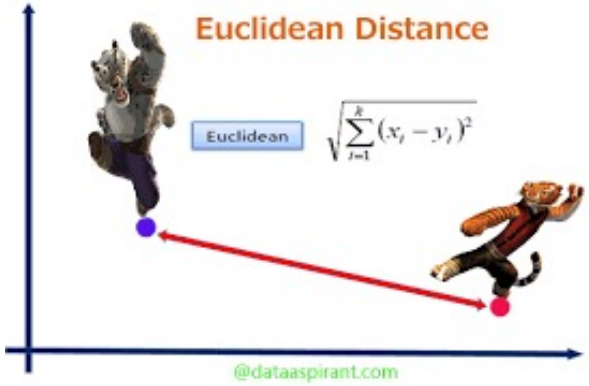
$D(u,v) < D(u,k); e_{uv} \in E, e_{u,k} \notin E$

Projected Metric Embedding (KDD18)

- However, directly applying the Euclidean distance as a metric will be problematic !
  - Mathematically, It is geometrically restrictive and an ill-posed algebraic system.
  - On the other hand, one object may have multiple aspects.

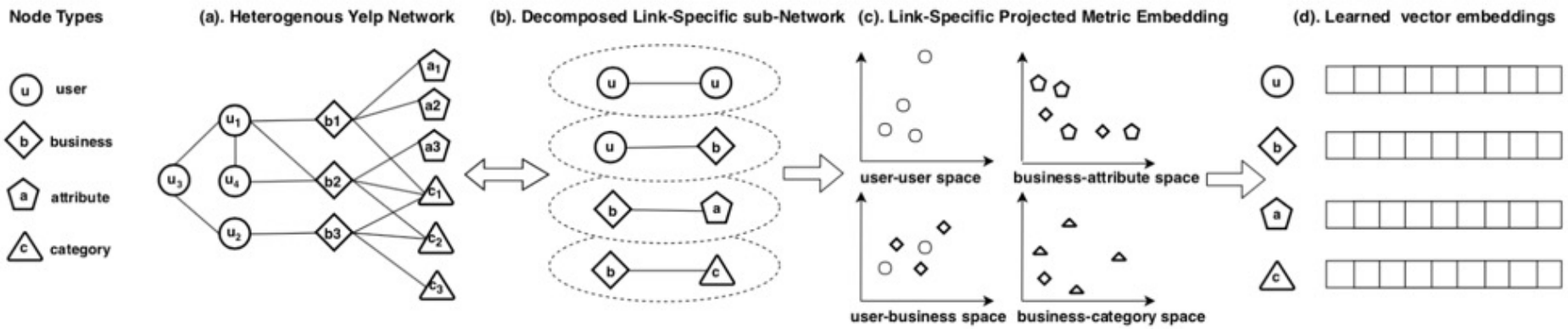
- To address these issues:
  - PME introduces relation-specific projection embedding matrices.
  - Model objects and relations in distinct spaces.
    - One shared object space.
    - Multiple relation spaces. (relation-specific object spaces).





Hence, it is possible that some objects are far away from each other in the object space, but are close to each other in the corresponding relation spaces.

Projected Metric Embedding (KDD18)



- Decompose the HIN to link-specific sub-networks.
- Metric learning on each link-specific space.

$$\mathbf{v}_\mathbf{i}^\mathbf{r} = \mathbf{M}_\mathbf{r}\mathbf{v}_\mathbf{i} \qquad d_r(v_i, v_j) = \|\mathbf{M}_\mathbf{r}\mathbf{v}_\mathbf{i} - \mathbf{M}_\mathbf{r}\mathbf{v}_\mathbf{j}\|, r \in \mathcal{R}$$

- Learn shared node embeddings and relation-specific space.

$$\min_{\mathbf{v}_*, \mathbf{M}_*} \sum_{r \in \mathcal{R}} \sum_{(v_i, v_j) \in D_r} \sum_{(v_i, v_k) \notin D_r} [m + f_r(v_i, v_j)^2 - f_r(v_i, v_k)^2]_+$$
$$s.t. \quad \|\mathbf{v}_*\| <= 1 \quad and \quad \|\mathbf{M}_*\| <= 1$$

Projected Metric Embedding (KDD18)

Model optimization:

- Bi-directional Negative Sampling Strategy

$$\min_{\mathbf{v}_*, \mathbf{M}_*} \sum_{r \in \mathcal{R}} \sum_{(v_i, v_j) \in D_r} \sum_{(v_i, v_k) \notin D_r} [m + f_r(v_i, v_j)^2 - f_r(v_i, v_k)^2]_+$$
$$s.t. \quad \|\mathbf{v}_*\| <= 1 \quad and \quad \|\mathbf{M}_*\| <= 1$$

- Directly optimizing this equation is expensive!

  Inspired by the negative sampling techniques

$$O = \sum_{r \in \mathcal{R}} \sum_{(v_i, v_j) \in D_r} \Big(\sum_{k=1}^{K} E_{v_k} \sim p_n(v) [m + f_r(v_i, v_j)^2 - \boxed{f_r(v_i, v_k)^2}]_+ + \sum_{k=1}^{K} E_{v_k} \sim p_n(v) [m + f_r(v_i, v_j)^2 - \boxed{f_r(v_k, v_j)^2}]_+ \Big)$$

first fix vertex v_i and edge type r, then generate K negative vertices v_k

then fix right side of e_{ijr} , and sample K negative vertex from the left side

Projected Metric Embedding (KDD18)

Model optimization:

- ❑ Loss-aware Adaptive Positive Sampling Strategy

1. **A sequence of the losses for each sub-network.**

$$L = (l_1, l_2, l_3, \ldots, l_{\|\mathcal{R}\|})$$

2. **Simply calculate the sum of the losses.**

$$L_{sum} = \sum_{r \in \mathcal{R}} l_r$$

3. **Draw a random value within the range of [0,1].**

$$x \sim Uniform(0,1)$$

4. **To see which interval the random fails into.**

$$[\sum_{j=0}^{r-1} \frac{l_j}{L_{sum}}, \sum_{j=0}^{r} \frac{l_j}{L_{sum}})$$

Algorithm 1 Training PME model

Input: A heterogeneous network $G(V, E, W, \mathcal{R})$, number of stochastic gradient steps, N, number of negative samples for each positive sample, K;

Output: Embeddings for network vertices and relation-specific projection matrix. (i.e., $\mathbf{v}, M_r$);

```
iter ← 0 :
while iter < N do
    if iter = 0 then
        Initialize the positive sampling probability as proportional to the original link distribution from G;
    else
        Sample M positive examples based on adaptive positive sampling strategy;
    End if
    For each sampled positive edge, sample K negative vertices from both sides of the edge;
    Compute gradients and update parameters;
    Censor the norm of v and projection matrix M_r;
    Compute relation-specific subgraph loss, and update the positive sampling probability;
    iter ← iter + 1 :
end
```

Projected Metric Embedding (KDD18)

Experimental Results

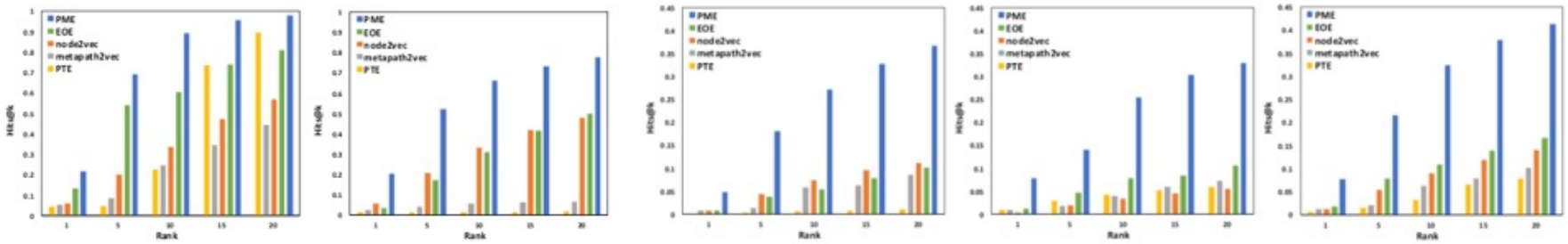
- ❖ Binary Link Classification (Yelp challenge dataset)

Table 4: AUC scores on NV network

	PME	node2vec	PTE	EOE	metapath2vec
Overall	**0.9618**	0.8789	0.7494	0.8562	0.6232
user-user	**0.9672**	0.8909	0.6347	0.9033	0.5141
user-business	**0.9590**	0.8835	0.8615	0.9129	0.8179
business-attribute	**0.9376**	0.7522	0.8944	0.9201	0.5653
business-category	**0.9896**	0.9233	0.9652	0.9819	0.7725

- ❖ Prediction accuracy (Hit ratio)

$$Hits@k = \frac{\#hit@k}{\|D^{+}_{test}\|}$$



(a) business-attribute (b) business-category (c) user-user (d) user-business (e) overall

2. Reasoning for Graph-based Recommendation

Outline

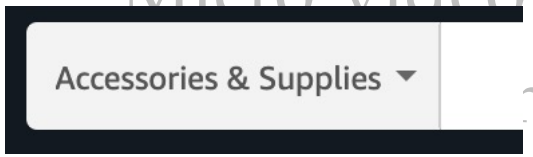
- 1 Introduction
- 2 Preliminary of Graph-based Explainable Recommendation
- 3 Literature Review of Graph-based Explainable Recommendation
- 4 Conclusion

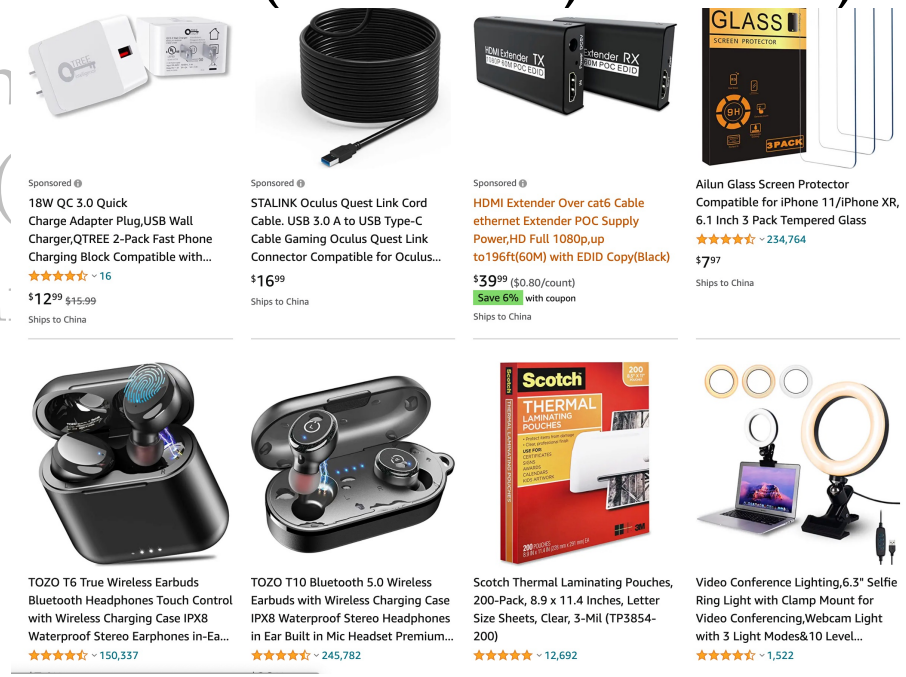
1 Introduction

- Exploded products/stores on e-commerce

1 Introduction

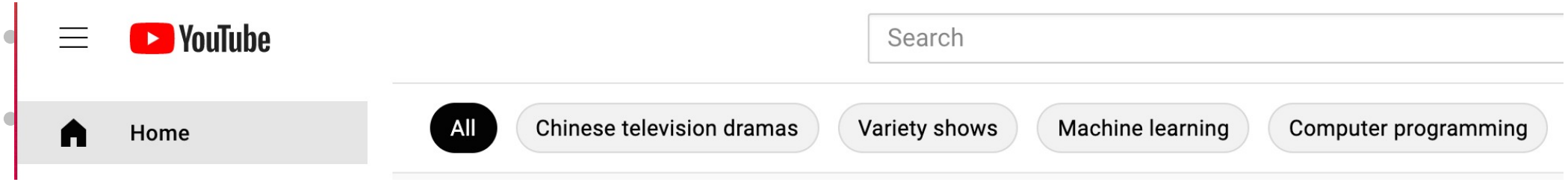
- The application scenario of recommendation:
- E-commerce recommendation (Amazon, Taobao, etc.)
- Micro-video recommen
- Point-of-interest recom





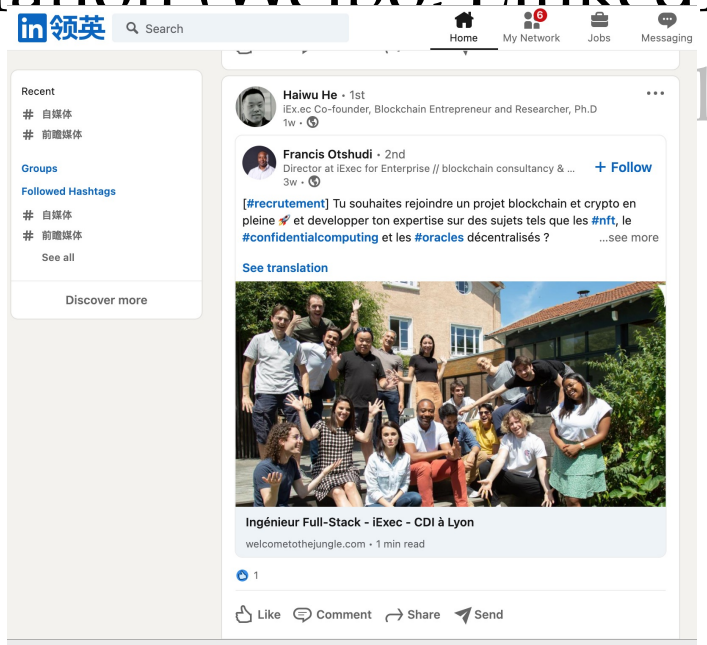
1 Introduction

- The application scenario of recommendation:
- E-commerce recommendation (Amazon, Taobao, etc.)
- Micro-video recommendation (YouTube, Tiktok, etc.)



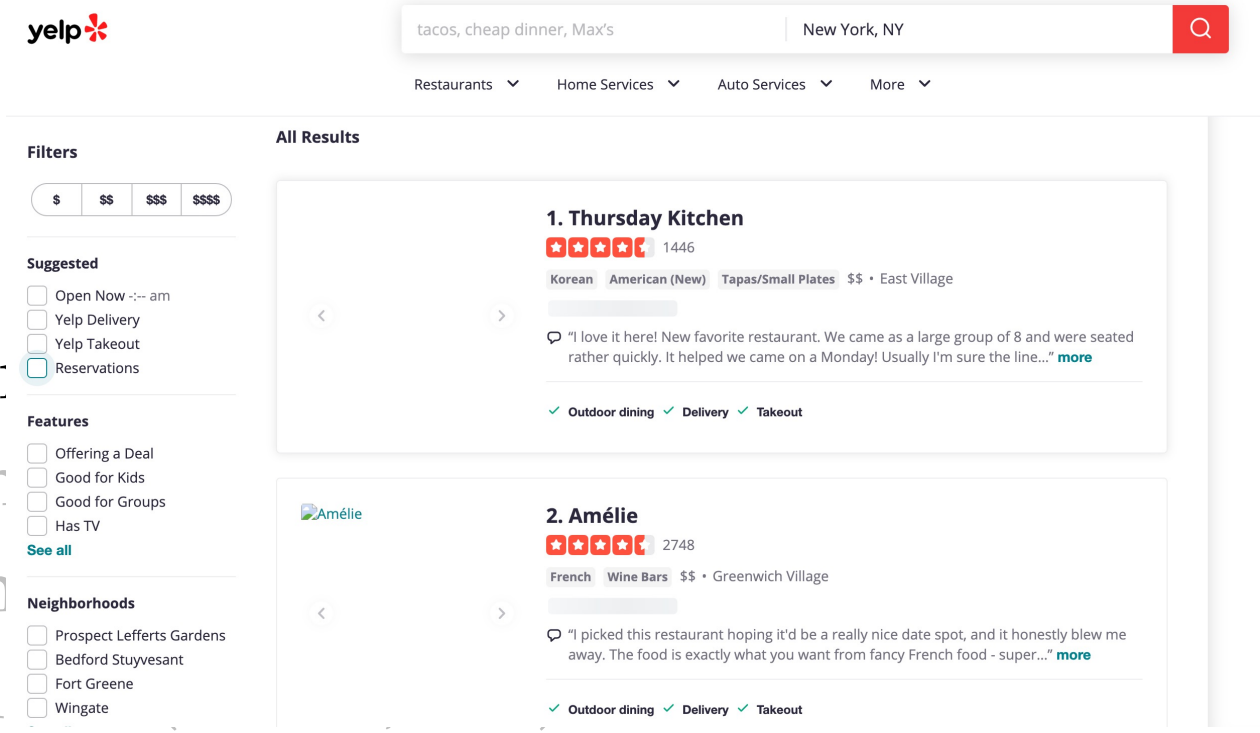
1 Introduction

- The application scenario of recommendation:
- E-commerce recommendation (Amazon, Taobao, etc.)
- Micro-video recommendation (YouTube, Tiktok, etc.)
- Post recommendation (Weibo, LinkedIn, etc.)
- Point-of-interest lp, etc.)



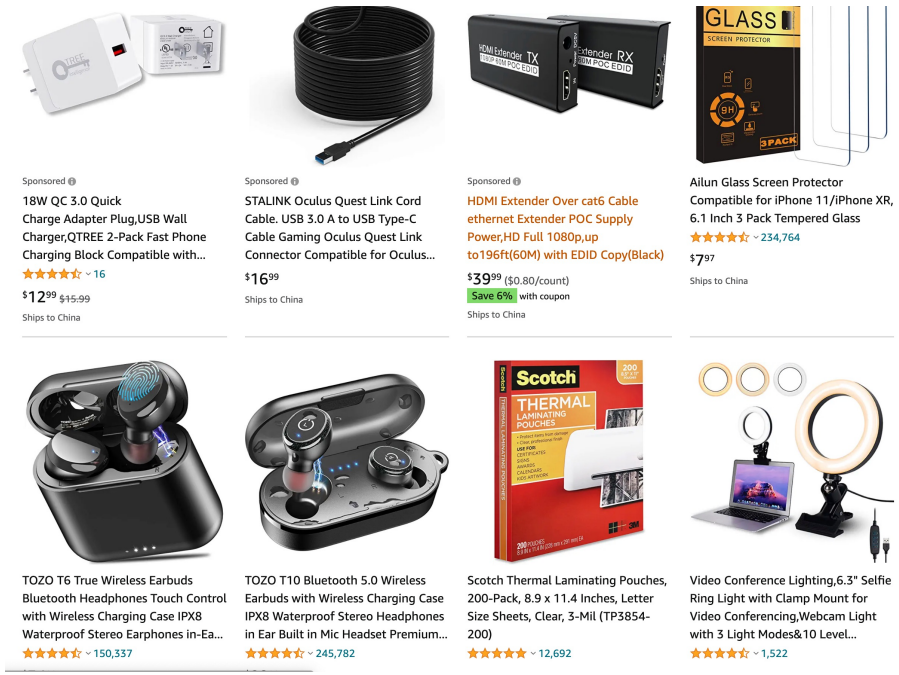
1 Introduction

- The application sce
- E-commerce recor
- Micro-video recon
- Post recommenda
- Point-of-interest recommendation (Yelp, etc.)



1 Introduction

- Why is the recommendation systems important?
  - For the merchant:
    - Bring more earnings through ads
    - Always catching attention and stream (promotion)
  - For the customers:
    - Saving time to choose needed products



1 Introduction

- But how to make the recommendation believable ?
  - Improve the performance of the recommendation's result
  - Improve the transparency of the recommendation

- Explainable recommendation
  - They not only provide users or system designers with recommendation results, but also explanations to clarify why such items are recommended.

1 Introduction

- Why is the explainable recommendation important? [1]
  - Avoid black box
  - Give more persuasive, transparent, trustworthy recommendations
  - Facilitate system designers to diagnose, debug and refine the recommendation algorithm.

[1] Zhang Y, Chen X. Explainable recommendation: A survey and new perspectives[J]. arXiv preprint arXiv:1804.11192, 2018.

1 Introduction

- Early explainable recommendation
  - Naïve model
    - Based on collaborative filtering [1]
    - Based on content [2]
    - Based on rule mining [3]

  - Advantages: Good explanation
  - Disadvantages: Low performance of recommendation

[1] Abdollahi B, Nasraoui O. Explainable matrix factorization for collaborative filtering[C]//WWW. 2016: 5-6.
[2] Zanon A L, Souza L, Pressato D, et al. WordRecommender: An Explainable Content-Based Algorithm based on Sentiment Analysis and Semantic Similarity[C]//Proceedings of the Brazilian Symposium on Multimedia and the Web. 2020: 181-184.
[3] Ma W, Zhang M, Cao Y, et al. Jointly learning explainable rules for recommendation with knowledge graph[C]// WWW. 2019: 1210-1221.

1 Introduction

- Later, explainable recommendation
  - Based on interpretable machine learning
    - Factorization Models [1]
    - Topic model [2]
    - …
  - Advantages: Better performance of recommendation than earlier methods
  - Disadvantages: Focus more on reviews and tend to ignore side information.

[1] Liu N, Ge Y, Li L, et al. Explainable recommender systems via resolving learning representations[C]//Proceedings of the 29th ACM International Conference on Information & Knowledge Management. 2020: 895-904.
[2] Ren Z, Liang S, Li P, et al. Social collaborative viewpoint regression with explainable recommendations[C]//WSDM. 2017: 485-494.

1 Introduction

- Currently, explainable recommendation
  - Based on graph neural networks
    - Involves social information [1]
    - Involves attributes information [2]

  - Advantages: State-of-the-art performance of recommendation's result
  - Disadvantages: Worse explainability than the earlier methods

[1] Wang S, Hu L, Wang Y, et al. Graph learning approaches to recommender systems: A review[J]. arXiv preprint arXiv:2004.11718, 2020.
[2] Samih A, Adadi A, Berrada M. Towards a knowledge based explainable recommender systems[C]//Proceedings of the 4th International Conference on Big Data and Internet of Things. 2019: 1-5.

1 Introduction

- Therefore, there are more and more researches focusing on exploring the explainability of graph-based recommendations.

- Tutorial Structure
  - Introduce the premilitary of graph-based explainable recommendation,
  - Summarize and classify the literature review of graph-based explainable recommendation, and
  - Provide potential future directions.

Outline

- 1 Introduction
- 2 Preliminary of Graph-based Explainable Recommendation
- 3 Literature Review of Graph-based Explainable Recommendation
- 4 Conclusion

2 Preliminary

- Definition of Graph-based recommendation
  - Assuming a graph $G = (V, E)$, each edge $e \in E$, and each entity $v \in V$, e represents a particular relation r of two entities (v_1, v_2) linked by edge e. Each entity v belongs to a particular type t $\in T$. On the recommendation graph G, if ignoring side information, types $T = \{U, I\}$, where U and I mean the user and item sets, respectively. If considering attributes of items A, types $T = \{U, I, A\}$.

2 Preliminary

- Problem formulation of Graph-based explainable recommendation
  - Given the user set $U = \{u_1, u_2, \dots, u_n\}$, the each user u_n 's historical item set $I_n = \{i_1, i_2, \dots, i_m\}$, and each item i_m 's related attribute set $A_m = \{a_1, a_2, \dots, a_p\}$, the graph-based explainable recommendation is to predict the next q items $\{i_{m+1}, i_{m+2}, i_{m+q}\}$ associated to each user u_n, according to the given I_n and related attributes $\{A_1, A_2, \dots, A_m\}$. At the same time, give the explanation why predict the next q items.

2 Preliminary

- Graph-based explainable recommendation

Input:
- Users
- Each user's historical items
- Each item's attributes

Explainable Recommender Systems

Output:
- Predicted item(s) for each user
- Explanation of each predicted item

2 Preliminary

- Evaluation of Graph-based explainable recommendation
  - Performance of recommendation results
  - Quality of explanation

2 Preliminary

- Evaluation of Graph-based explainable recommendation
  - Performance of recommendation results
    - Hit Ratio (HR) @ K
    - Normalized Discounted Cumulative Gain (NDCG) @ K
    - Precision @ K
    - Recall @ K

2 Preliminary

- Evaluation of Graph-based explainable recommendation
  - Performance of recommendation results
    - Hit Ratio (HR) @ K: Hit ratio (HR) is always used to measure the ratio of the recommendation system hit to the correct items, which reflects the recall quality.
    - Mathematically,

$$HR@K = \frac{\text{NumberofHits@K}}{|GT|}$$

where $|GT|$ is the nu neans the number of the correct predicted item. K is the uation is focusing on.

2 Preliminary

- Evaluation of Graph-based explainable recommendation
  - Performance of recommendation results
    - Normalized Discounted Cumulative Gain (NDCG) @ K: It calculates the cumulative result with a normalization. The higher the correct hitting rank, the higher is the score.
    - Mathematically,
- where Z_k is the norm $NDCG@K = Z_k \sum_{i=1}^{K} \frac{2^{r_i} - 1}{\log_2(i+1)}$ rameter. If the $i - th$ predicted item is in t

2 Preliminary

- Evaluation of Graph-based explainable recommendation
  - Performance of recommendation results
    - Precision @ K: It is the ratio of predicted correctly in the top k prediction sets.
    - Mathematically,
- $$P = \frac{1}{n} \sum_{u \in \mathcal{U}} \frac{|L_u \cap B_u|}{|L_u|}$$

  where L_u and B_u repres truncated) and the correct prediction set, respectively.

2 Preliminary

- Evaluation of Graph-based explainable recommendation
  - Performance of recommendation results
    - Recall @ K: It is the ratio of predicted correctly in the top k test set.
    - Mathematically,
- $$R = \frac{1}{n} \sum_{u \in \mathcal{U}} \frac{|L_u \cap B_u|}{|B_u|}$$

  where L_u and B_u repres truncated) and the correct prediction set, respectiv

2 Preliminary

- Evaluation of Graph-based explainable recommendation
  - Performance of recommendation results
  - Quality of explanation

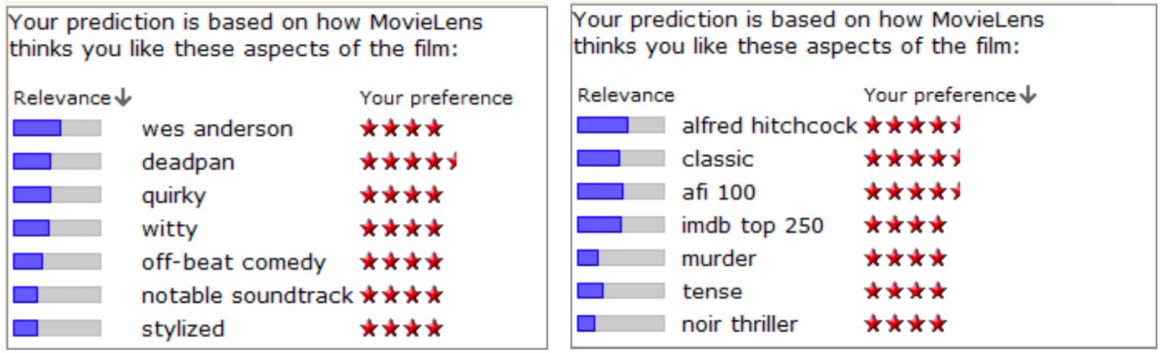
2 Preliminary

- Evaluation of Graph-based explainable recommendation [1]
  - Quality of explanation
    - User study
    - Online evaluation
    - Offline evaluation
    - Case study

[1] Zhang Y, Chen X. Explainable recommendation: A survey and new perspectives[J]. arXiv preprint arXiv:1804.11192, 2018.

2 Preliminary

- Evaluation of Graph-based explainable recommendation [1]
  - Quality of explanation
    - User study
      - The study will design some questions or tasks for the subjects to answer or complete, and conclusions will be derived from the responses of the subjects.



[2]

[1] Zhang Y, Chen X. Explainable recommendation: A survey and new perspectives[J]. arXiv preprint arXiv:1804.11192, 2018.
[2] Vig, J., S. Sen, and J. Riedl. 2009. “Tagsplanations: explaining recom- mendations using tags”. In: Proceedings of the 14th international conference on Intelligent user interfaces. ACM. 47–56.

2 Preliminary

- Evaluation of Graph-based explainable recommendation [1]
  - Quality of explanation
    - Online evaluation
      - See if the explanations can help to make users accept the recommendations. It usually tests online. [2]
      - E.g.: On a commercial web browser, the experimental group receives testing explanations, the comparison group receives the baseline ‘People also viewed’ explanations, and a control group that receives no explanation. The click-through rate of each group is calculated to evaluate the effect of providing personalized explanations.

[1] Zhang Y, Chen X. Explainable recommendation: A survey and new perspectives[J]. arXiv preprint arXiv:1804.11192, 2018.
[2] Zhang, Y., G. Lai, M. Zhang, Y. Zhang, Y. Liu, and S. Ma. “Explicit factor models for explainable recommendation based on phrase-level sentiment analysis”. SIGIR. ACM. 83–92. 2014.

2 Preliminary

- Evaluation of Graph-based explainable recommendation [1]
  - Quality of explanation
    - Offline evaluation
      - Model Fidelity (MF)

$$Model\ Fidelity = \frac{|explainable\ items \cap recommended\ items|}{|recommended\ items|}$$

[1] Zhang Y, Chen X. Explainable recommendation: A survey and new perspectives[J]. arXiv preprint arXiv:1804.11192, 2018.

2 Preliminary

- Evaluation of Graph-based explainable recommendation [1]
  - Quality of explanation
    - Offline evaluation
      - Bilingual Evaluation Understudy (BLEU) [2], *if sentence*

$$p_n = \frac{\sum_{C \in \{Candidates\}} \sum_{n\text{-}gram \in C} Count_{clip}(n\text{-}gram)}{\sum_{C' \in \{Candidates\}} \sum_{n\text{-}gram' \in C'} Count(n\text{-}gram')}.$$

$Candidates$ are the set of generated sentences.

$$\mathrm{BP} = \begin{cases} 1 & \text{if } c > r \\ e^{(1-r/c)} & \text{if } c \leq r \end{cases}.$$

c is the length of the generated sentence and r is the ground-truth sentence length.

Then,

$$\mathrm{BLEU} = \mathrm{BP} \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right).$$

w_n is a positive weights

[1] Zhang Y, Chen X. Explainable recommendation: A survey and new perspectives[J]. arXiv preprint arXiv:1804.11192, 2018.
[2] Papineni K, Roukos S, Ward T, et al. Bleu: a method for automatic evaluation of machine translation[C]//ACL. 2002: 311-318.

2 Preliminary

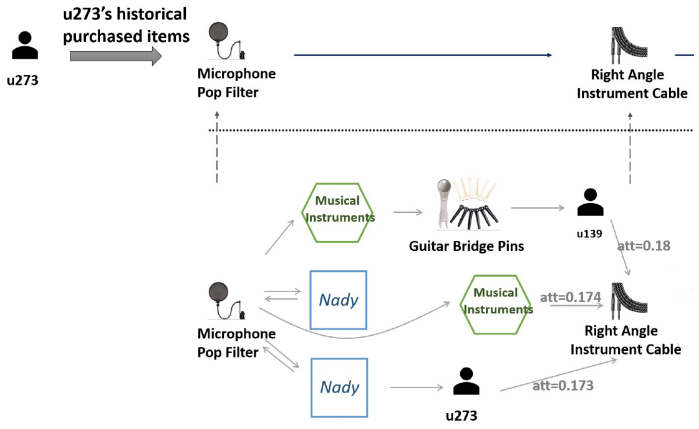
- Evaluation of Graph-based explainable recommendation [1]
  - Quality of explanation
    - Offline evaluation
      - Recall-oriented Understudy for Gisting Evaluation (ROUGE), *if sentence*

$$\text{ROUGE-N} = \frac{\sum_{S \in \{ReferemceSummaries\}} \sum_{gram_n \in S} Count_{match}(gram_n)}{\sum_{S \in \{ReferenceSummaries\}} \sum_{gram_n \in S} Count(gram_n)} \quad (1)$$

[1] Zhang Y, Chen X. Explainable recommendation: A survey and new perspectives[J]. arXiv preprint arXiv:1804.11192, 2018.
[2] Lin C Y. Rouge: A package for automatic evaluation of summaries[C]//Text summarization branches out. 2004: 74-81.

2 Preliminary

- Evaluation of Graph-based explainable recommendation [1]
  - Quality of explanation
    - Case study
      - Providing case studies can help to understand the intuition behind the explainable recommendation model and the effectiveness of explanations.



[2]

[1] Zhang Y, Chen X. Explainable recommendation: A survey and new perspectives[J]. arXiv preprint arXiv:1804.11192, 2018.
[2] Chen H, Li Y, Sun X, et al. Temporal meta-path guided explainable recommendation[C]//WSDM. 2021: 1056-1064.

Outline

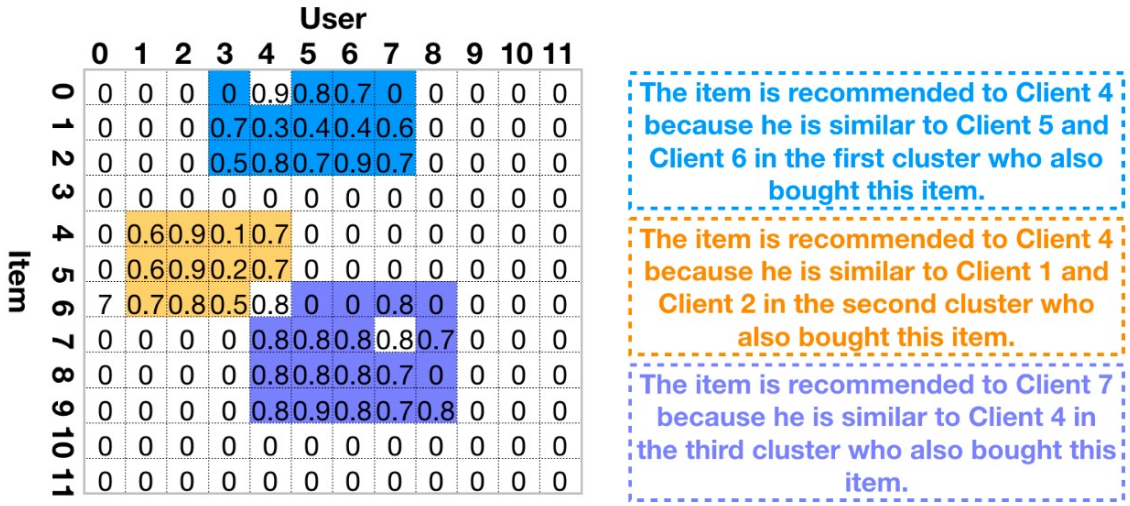
- 1 Introduction
- 2 Preliminary of Graph-based Explainable Recommendation
- 3 Literature Review of Graph-based Explainable Recommendation
- 4 Conclusion

3 Literature Review

- Classification of graph-based explainable recommendation
  - Homogeneous graph
    - Regard collaborative information as reasoning
  - Heterogeneous graph
    - Regard path as reasoning
    - Generate sentences as reasoning
    - Extract sentences as reasoning
    - Consider visualized relation as reasoning
    - …

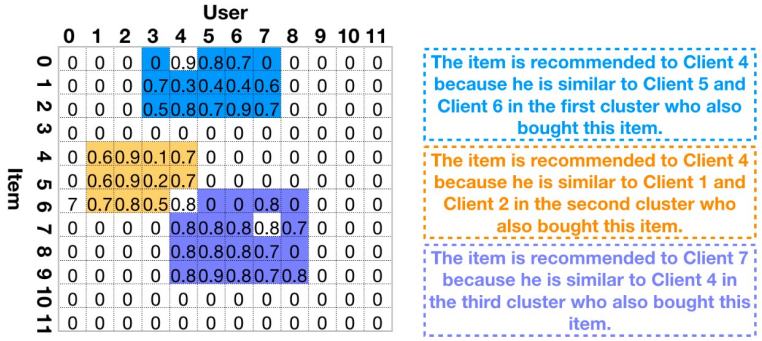
3 Literature Review

- Classification of graph-based explainable recommendation
  - On homogeneous graph
    - A bipartite graph is constructed on the recommendation data (only including users and items).
    - Feature: The explanation is mostly based on the collaborative information.



3 Literature Review

- Classification of graph-based explainable recommendation
  - On homogeneous graph – regard collaborative information as reasoning
  - Related work [1]
    - If a user-item pair falls into multiple co-clusters, it can thus generate multiple user-based and item-based explanations from each of the co-cluster.
    - Shortcoming: It does not involve the external information.



[1] Heckel R, Vlachos M, Parnell T, et al. Scalable and interpretable product recommendations via overlapping co-clustering[C]//ICDE. IEEE, 2017: 1033-1044.

3 Literature Review

- Classification of graph-based explainable recommendation
  - Homogeneous graph
    - Regard collaborative information as reasoning
  - Heterogeneous graph
    - Regard path as reasoning
    - Regard words as reasoning
    - Regard sentences as reasoning
    - Regard visual graph as reasoning
    - …

3 Literature Review

- Classification of graph-based explainable recommendation
  - On heterogeneous graph
    - A heterogenous graph is constructed on the recommendation data (including users, items and other external information).
    - Feature: It involves the external information compared with the homogeneous graph-based explainable recommendation. Moreover, the explanation tends to be more kinds.

3 Literature Review

- Classification of graph-based explainable recommendation
  - On heterogeneous graph - Regard path as reasoning
    - It is a naïve idea to explain the recommendation on the graph. Intuitively, paths with lengths 2 on the graph represents the relation between two nodes and paths with more than lengths 2 represent compositional relations.
    - Meta-path [1]
    - Meta-path instance [2]

[1] Sun Y, Han J. Mining heterogeneous information networks: principles and methodologies[J]. Synthesis Lectures on Data Mining and Knowledge Discovery, 2012, 3(2): 1-159.
[2] Sun Y, Han J. Meta-path-based search and mining in heterogeneous information networks[J]. Tsinghua Science and Technology, 2013, 18(4): 329-338.

3 Literature Review

- Classification of graph-based explainable recommendation
  - On heterogeneous graph - Regard path as reasoning
    - Meta-path
      - A meta path is an ordered sequence of node types and edge types defined on the network schema, which describes a composite relation between the nodes' types involved.
      - e.g., Author-Paper-Author (APA) → co-authors;
      - Author-Paper-Venue-Paper-Author (APVPA) →two authors who published papers in the same venue.

3 Literature Review

- Classification of graph-based explainable recommendation
  - On heterogeneous graph - Regard path as reasoning
    - Meta-path
    - Related work [1]
      - Each meta-path represents a semantic meaning in the model.

Table 1: The meanings and corresponding recommendation models of meta paths.

No.	Meta Path	Semantic Meaning	Recommendation Model
1	UU	friends of the target user	Social recommendation
2	UGU	users in the same group of the target user	Member recommendation
3	UMU	users who view the same movies with the target user	Collaborative recommendation
4	UMTMU	users who view the movies having the same types with that of the target user	Content recommendation

[1] Shi C, Zhang Z, Luo P, et al. Semantic path based personalized recommendation on weighted heterogeneous information networks[C]//CIKM. 2015: 453-462.

3 Literature Review

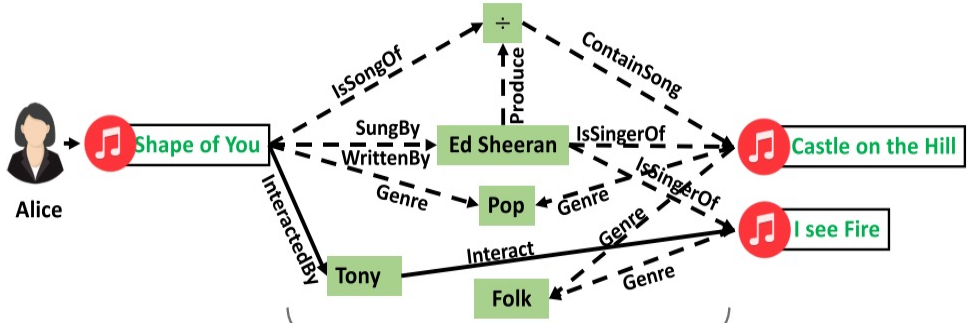
- Classification of graph-based explainable recommendation
  - On heterogeneous graph - Regard path as reasoning
    - Meta-path
    - Related work
      - Shortcoming: Meta-path schema can only provide general and high-level explanations, because each node in the meta-path represents a type.

Table 1: The meanings and corresponding recommendation models of meta paths.

No.	Meta Path	Semantic Meaning	Recommendation Model
1	UU	friends of the target user	Social recommendation
2	UGU	users in the same group of the target user	Member recommendation
3	UMU	users who view the same movies with the target user	Collaborative recommendation
4	UMTMU	users who view the movies having the same types with that of the target user	Content recommendation

3 Literature Review

- Classification of graph-based explainable recommendation
  - On heterogeneous graph - Regard path as reasoning
    - Meta-path instance
      - A meta path instance context is the general information of several instances from the same start node to the same end node.
      - e.g., All meta path instances from **Alice** to I see Fire : [1]
      - ϕ_1: **Alice** →Shape of You →**Ed Sheeran** → I see Fire
      - ϕ_2 : **Alice** → Shape of You → **Tony** → I see Fire
      - **A meta path instance context from Alice to I see Fire** : $\phi=\alpha$_1 ϕ_1+α_2 ϕ_2

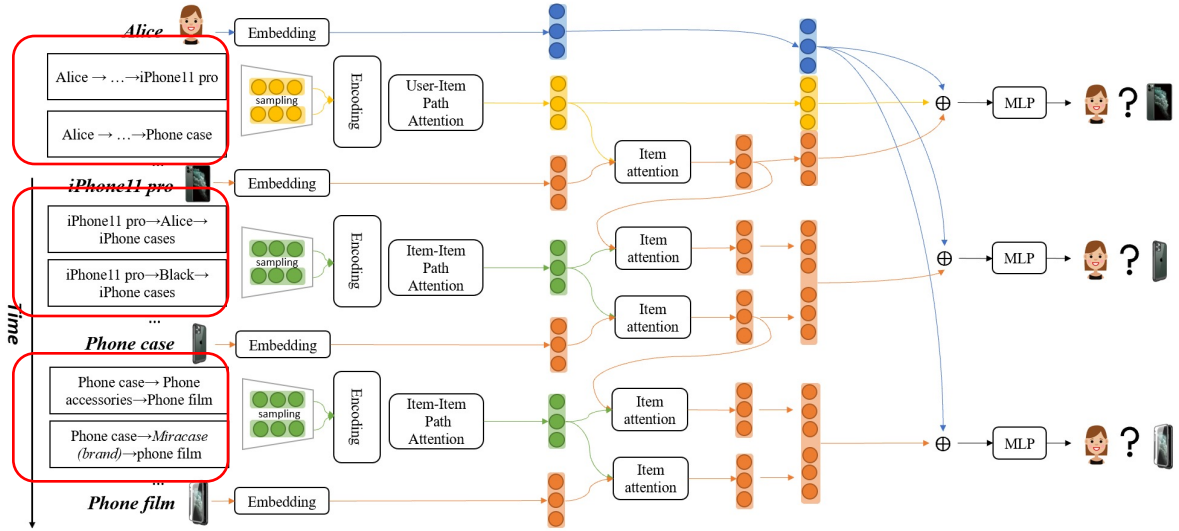


[1] Hu B, et al. Leveraging meta-path based context for top-n recommendation with a neural co-attention model[C]//KDD. 2018: 1531-1540.

3 Literature Review

- Classification of graph-based explainable recommendation
  - On heterogeneous graph - Regard path as reasoning
    - Meta-path instance
    - Related work (TMER) [1]

Meta-path instance as explanation



[1] Chen H, Li Y, Sun X, et al. Temporal meta-path guided explainable recommendation[C]//WSDM. 2021: 1056-1064.

3 Literature Review

- Classification of graph-based explainable recommendation
  - On heterogeneous graph - Regard path as reasoning
    - Meta-path instance
    - Related work (TMER) [1]
      - loss function:

$$loss_{u,i} = -E_{j \sim \boxed{P_{neg}}} \left[\log \left(\boxed{1} - \boxed{r_{u,j}} \right) \right]$$

Noise Distribution

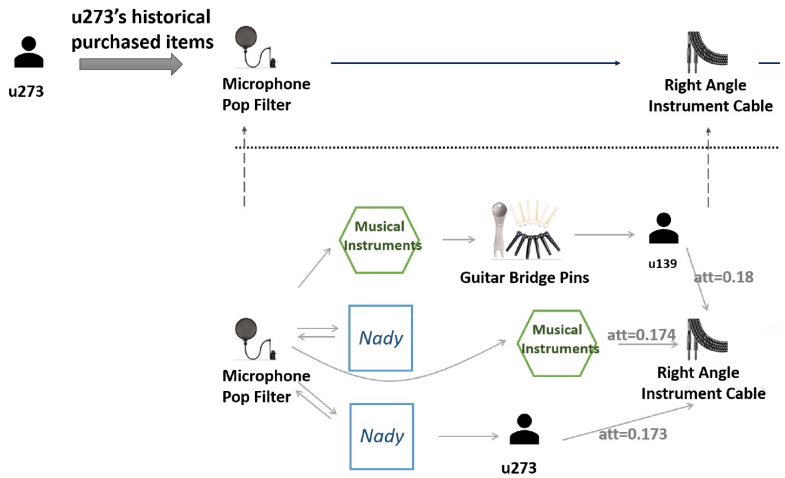
Positive rating

Calculated Rating of user u and negative item j

[1] Chen H, Li Y, Sun X, et al. Temporal meta-path guided explainable recommendation[C]//WSDM. 2021: 1056-1064.

3 Literature Review

- Classification of graph-based explainable recommendation
  - On heterogeneous graph - Regard path as reasoning
    - Meta-path instance
    - Related work (TMER) [1]
      - Evaluation of explainability:

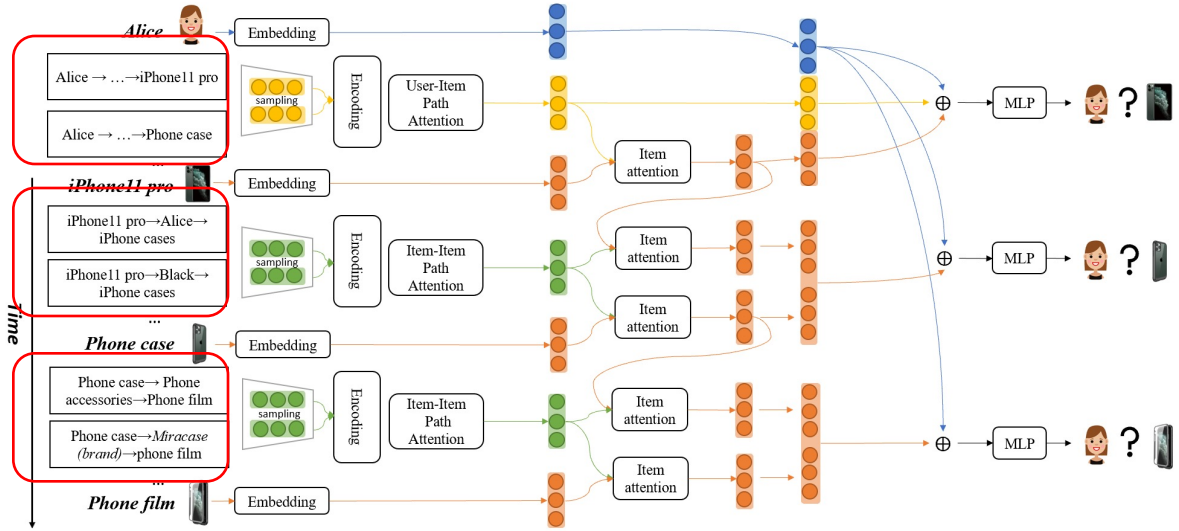


[1] Chen H, Li Y, Sun X, et al. Temporal meta-path guided explainable recommendation[C]//WSDM. 2021: 1056-1064.

3 Literature Review

- Classification of graph-based explainable recommendation
  - On heterogeneous graph - Regard path as reasoning
    - Meta-path instance
    - Related work (TMER) [1]

Meta-path instance as explanation

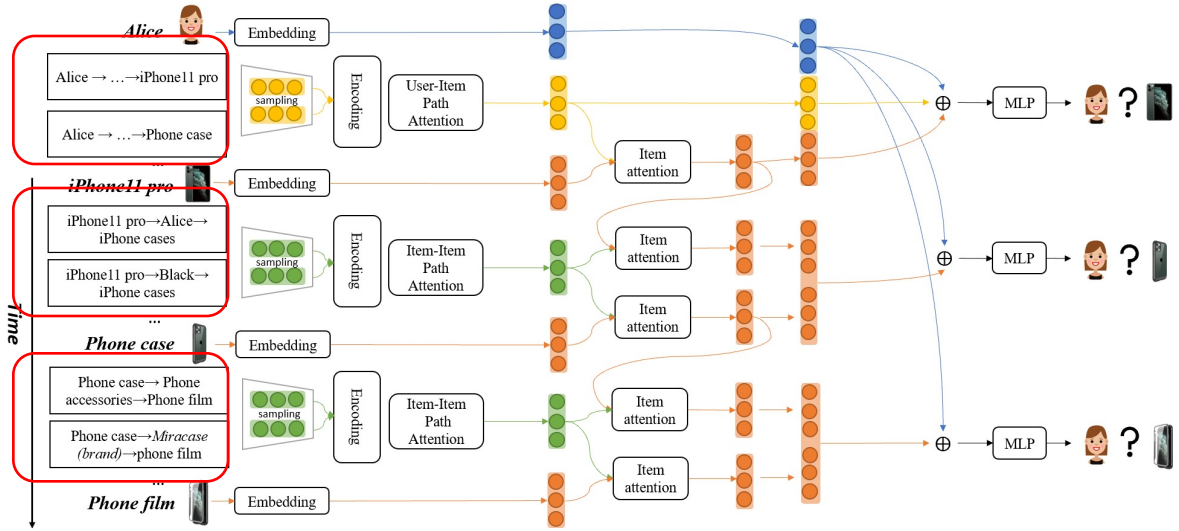


[1] Chen H, Li Y, Sun X, et al. Temporal meta-path guided explainable recommendation[C]//WSDM. 2021: 1056-1064.

3 Literature Review

- Classification of graph-based explainable recommendation
  - On heterogeneous graph - Regard path as reasoning
    - Meta-path instance
    - Related work (TMER) [1]

Meta-path instance as explanation



[1] Chen H, Li Y, Sun X, et al. Temporal meta-path guided explainable recommendation[C]//WSDM. 2021: 1056-1064.

3 Literature Review

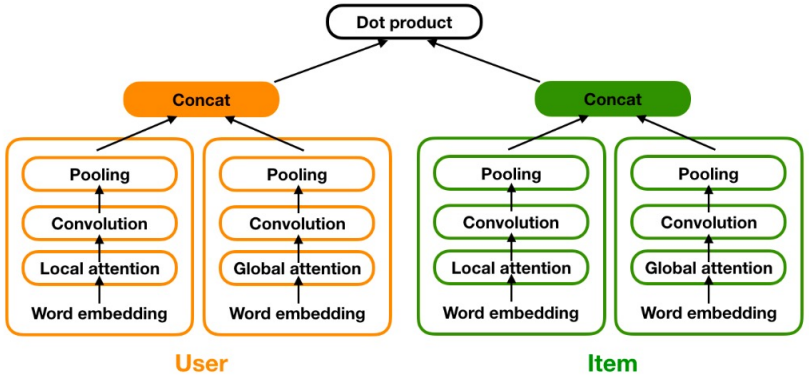
- Classification of graph-based explainable recommendation
  - On heterogeneous graph - Regard path as reasoning
  - Shortcoming:
    - Lower readable than natural language by users (customers)
    - Only focusing on the relation among users and items
    - Ignore rich information in reviews given by users

3 Literature Review

- Classification of graph-based explainable recommendation
  - On heterogeneous graph - Regard words as reasoning
  - To avoid the previous shortcomings, using a more natural language way to explain the recommendation is a solution. One of the most naïve idea is to extract words in reviews as explanations.

3 Literature Review

- Classification of graph-based explainable recommendation
  - On heterogeneous graph - Regard words as reasoning
  - Related work [1]
    - It leans dual local and global attentions for explanation purposes. When predicting the user-item rating, the model selectively chooses review words with different attention weights. Finally, the words can be regarded as explanations.



Professional, caring and thorough who goes above and beyond.
What else could you wish for your dentist to be? I went to many other dentists before but didn't get such a professional approach as I got from dr. Nataly.
She was able to save my front tooth, while all the other dentists prompted me to take it out and put an implant or bridge.
Dr. Nataly made a real miracle and I am extremely thankful to her. Unlike other doctors, she is oriented on saving your own teeth.
Along with her magic hands, she puts you at ease and will not let you go until she reaches the best, desired result.

My entire family switched to Dr. Nataly and all of them are super happy!
Her entire team is as well wonderful.
Highly recommend Dr. Nataly to anybody!

[1] Seo S, Huang J, Yang H, et al. Interpretable convolutional neural networks with dual local and global attention for review rating prediction[C]//RecSys. 2017: 297-305.

3 Literature Review

- Classification of graph-based explainable recommendation
  - On heterogeneous graph - Regard words as reasoning
  - Shortcoming:
    - The idea is too naïve and is lower readable than natural language sentences by users (customers)

3 Literature Review

- Classification of graph-based explainable recommendation
  - On heterogeneous graph - Regard sentences as reasoning
  - To avoid the previous shortcomings, using natural language sentences to explain the recommendation is another way to provide explanations.

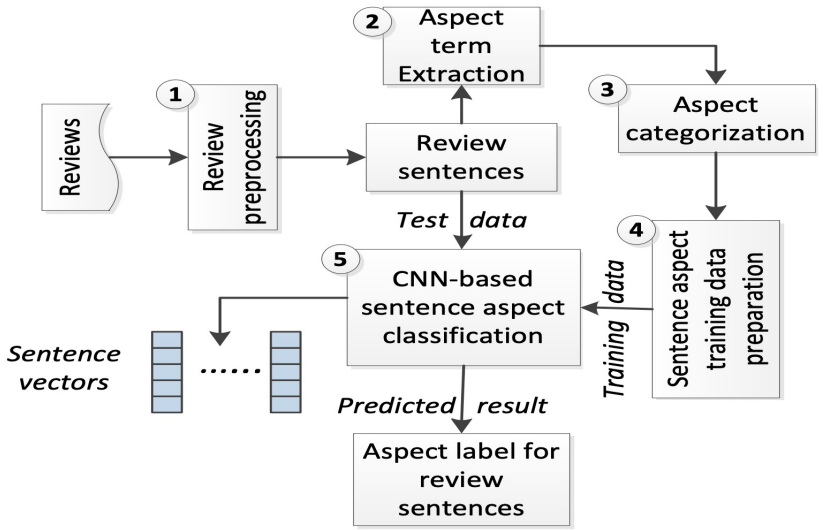
3 Literature Review

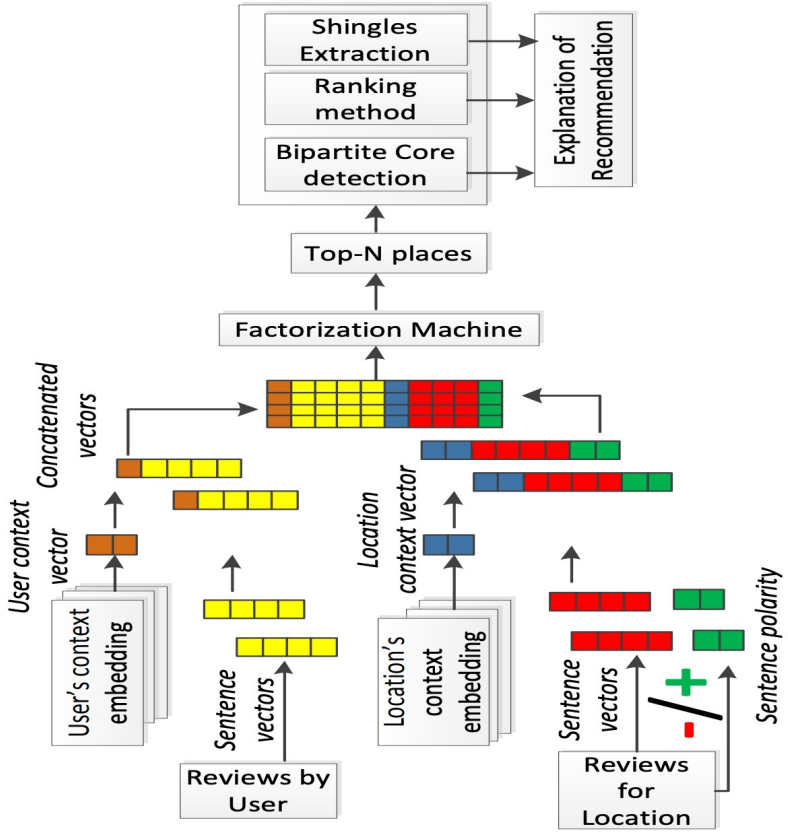
- Classification of graph-based explainable recommendation
  - On heterogeneous graph - Regard sentences as reasoning
  - Related work [1]
    - Point-of-Interest explainable recommendation research problem:
      - For the user u, and his historical visited places $l = \{l_1, l_2, \dots, l_n\}$, the model should give the recommended next place to go l_{n+1} and its explanation.

[1] Baral R, Zhu X L, Iyengar S S, et al. Reel: Review aware explanation of location recommendation[C]//Proceedings of the 26th conference on user modeling, adaptation and personalization. 2018: 23-32.

3 Literature Review

- Classification of graph-based explainable recommendation
  - On heterogeneous graph - Regard sentences as reasoning
  - Related work [1]





[1] Baral R, Zhu X L, Iyengar S S, et al. Reel: Review aware explanation of location recommendation[C]//Proceedings of the 26th conference on user modeling, adaptation and personalization. 2018: 23-32.

3 Literature Review

- Classification of graph-based explainable recommendation
  - On heterogeneous graph - Regard sentences as reasoning
  - Related work [1]
    - Shortcoming:
      - The explanation sentences are labeled/extracted from the input comments, which is not diverse.

[1] Baral R, Zhu X L, Iyengar S S, et al. Reel: Review aware explanation of location recommendation[C]//Proceedings of the 26th conference on user modeling, adaptation and personalization. 2018: 23-32.

3 Literature Review

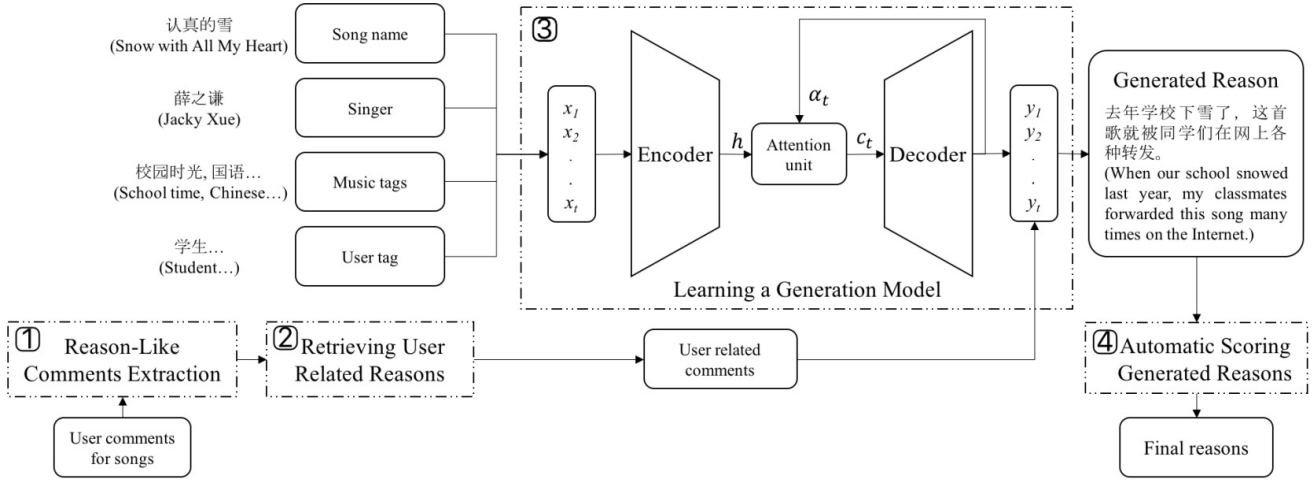
- Classification of graph-based explainable recommendation
  - On heterogeneous graph - Regard sentences as reasoning
  - Related work [1]
    - Music recommendation research problem: generate a reason as a sequence of words $Y = (y_1, y_2, \cdots, y_M)$ to explain why the user U should listen to the song S.

[1] Zhao G, Fu H, Song R, et al. Personalized reason generation for explainable song recommendation[J]. ACM Transactions on Intelligent Systems and Technology (TIST), 2019, 10(4): 1-21.

3 Literature Review

- Classification of graph-based explainable recommendation
  - On heterogeneous graph - Regard sentences as reasoning
  - Related work [1]

- Input:
  - users,
  - users' tag
  - songs related to each user,
  - tag, sords, singer related to each song
- Output:
  - Generated sentence as explanation for the song.



[1] Zhao G, Fu H, Song R, et al. Personalized reason generation for explainable song recommendation[J]. ACM Transactions on Intelligent Systems and Technology (TIST), 2019, 10(4): 1-21.

3 Literature Review

- Classification of graph-based explainable recommendation
  - On heterogeneous graph - Regard sentences as reasoning
  - Related work [1]
    - Explanation

[1] Zhao G, Fu H, Song R, et al. Personalized reason generation for explainable song recommendation[J]. ACM Transactions on Intelligent Systems and Technology (TIST), 2019, 10(4): 1-21.

3 Literature Review

- Classification of graph-based explainable recommendation
  - On heterogeneous graph - Regard sentences as reasoning
  - Related work [1]
    - Explanation

Input	User Tags	Generated Reason
Singer: 朴树 (Pushu) **Song name**: 我爱你，再见 (I love you but goodbye) **Tags**: 民谣 (Ballad) 大学时光 (College Years) 校园时光 (School Time) 光阴 (Time) 摇滚 (Rock)	学生，失恋 (Student, Lovelorn)	以前学校每天中午都会放这首歌，现在想想，我的初恋 (In the past, the school played this song at noon every day, now it is recalling my memory, my first love) (After breaking up, listening to this song, I feel drunk) 今天学校放了这首歌，我就知道这首歌是我的初恋 (As soon as the school played this song today, I know this song is my first love)
	晚睡，电音 (Sleep Late, Electronic Music)	晚上睡觉前听这首歌，越听越带感 (Listen to this song before going to bed. The more you listen, the higher you feel.) 半夜听这首歌，感觉好爽 (Listen to this song at midnight, I feel so high.)
	民谣，电音 (Ballad, Electronic Music)	每次听这首歌都会有一种震撼的感觉 (Every time I listen to this song, I have a very quiet feeling) 民谣的歌词都是摇滚的 (All the lyrics of ballad songs are rock and roll)
	失恋，晚睡 (Lovelorn, Sleep Late)	听完这初恋给我听的歌，现在听着听着就想哭了 (After listening to the song recommended by my first love, now I feel crying while listening it.) 听完这首歌就睡觉了，晚安 (Go to bed after listening to this song. Good Night)

[1] Zhao G, Fu H, Song R, et al. Personalized reason generation for explainable song recommendation[J]. ACM Transactions on Intelligent Systems and Technology (TIST), 2019, 10(4): 1-21.

3 Literature Review

- Classification of graph-based explainable recommendation
  - On heterogeneous graph - Regard sentences as reasoning
    - Shortcoming:
      - The sentence is sometimes not fluent because of the bottleneck of natural language processing method.

3 Literature Review

- Classification of graph-based explainable recommendation
  - On heterogeneous graph - Regard visual graph as reasoning
  - A visual interactive graph-based interface allows users to specify, refine and build item-preference profiles in a variety of domains. The interface facilitates expressions of taste through simple graph interactions and these preferences are used to compute personalized, fully transparent item recommendations for a target user.

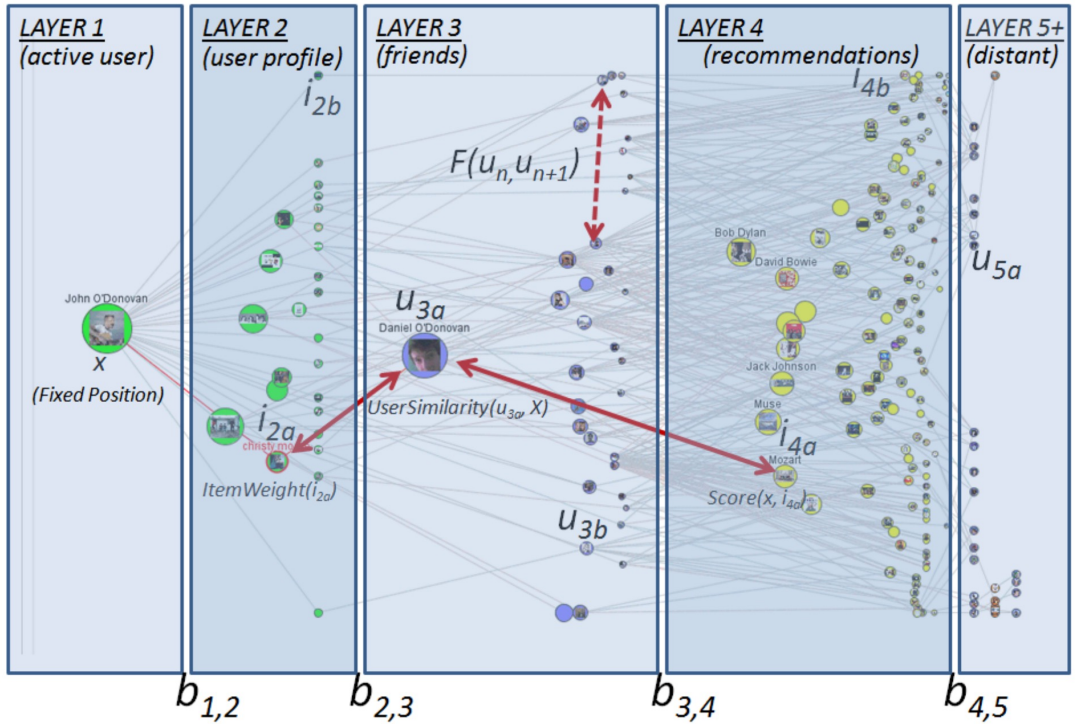
3 Literature Review

- Classification of graph-based explainable recommendation
  - On heterogeneous graph - Regard visual graph as reasoning
  - Related work [1]
    - The predictions are based on a collaborative analysis of preference data from a user's direct peer group on a social network. Users also learn a wealth of information about the preferences of their peers through interaction with our visualization.

[1] Gretarsson B, O'Donovan J, Bostandjiev S, et al. Smallworlds: visualizing social recommendations[C]//Computer graphics forum. Oxford, UK: Blackwell Publishing Ltd, 2010, 29(3): 833-842.

3 Literature Review

- Classification of graph-based explainable recommendation
  - On heterogeneous graph - Regard visual graph as reasoning
  - Related work [1]
    - The predictions are bas rom a user's direct peer group on a nation about the preferences of their pe



[1] Gretarsson B, O'Donovan J, Bostandjiev S, et al. Smallworlds: visualizing social recommendations[C]//Computer graphics forum. Oxford, UK: Blackwell Publishing Ltd, 2010, 29(3): 833-842.

3 Literature Review

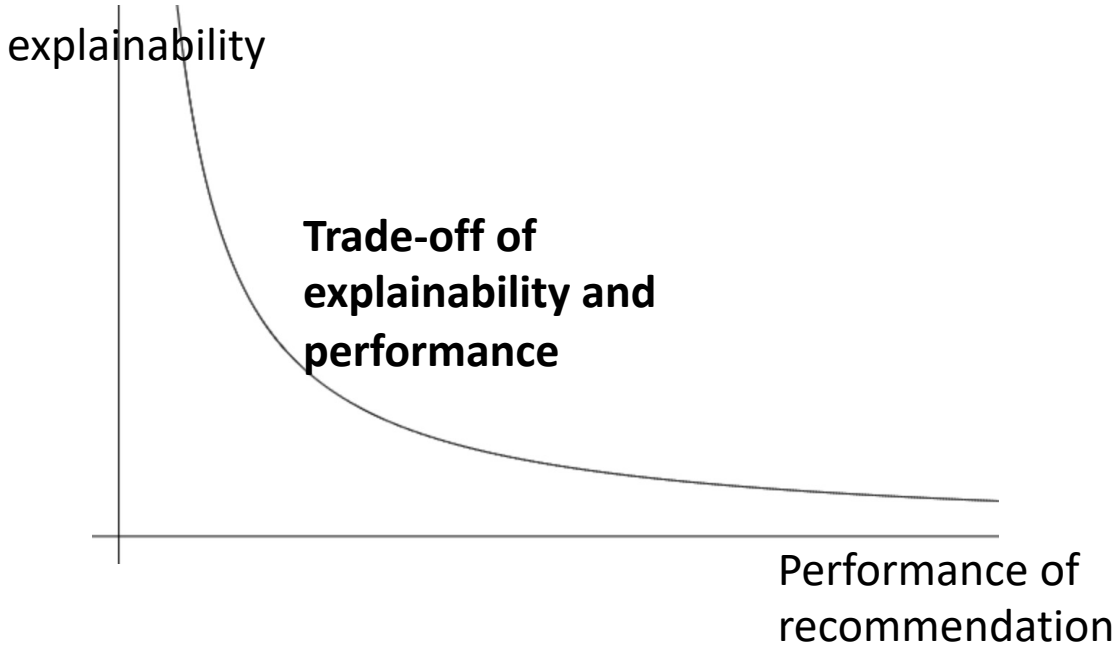
- Classification of graph-based explainable recommendation
  - On heterogeneous graph - Regard visual graph as reasoning
  - Shortcoming:
    - It is usually hard to provide enough contributions for the visual graph-based recommendation and they are coding-oriented. They adopt existing research work and is less novel with less contribution to the idea of providing explanations.

Outline

- 1 Introduction
- 2 Preliminary of Graph-based Explainable Recommendation
- 3 Literature Review of Graph-based Explainable Recommendation
- 4 Conclusion

4 Conclusion

- Introduce the recommendation system and its explainability
- Classify the explainable recommendation
  - Homogeneous graph
    - Regard collaborative information as reasoning
  - Heterogeneous graph
    - Regard path as reasoning
    - Regard words as reasoning
    - Regard sentences as reasoning
    - Regard visual graph as reasoning
    - …



4 Conclusion

- Future direction of graph-based explainable recommendation
  - Dynamic graph explainable recommendation
    - Sequential feature is an important information in recommendation and explanation. However, the existing graph-based explainable recommendation are almost based on static graphs.
    - Challenge:
      - How to design an *effective* and *efficient* dynamic graph model to do explainable recommendation?

4 Conclusion

- Future direction of graph-based explainable recommendation
  - Causality-based explainable recommendation
    - It has been adopted in machine learning to provide causal relation for better explainability in model.
    - Challenge:
      - How to combine the causal learning in graph to do recommendation?

4 Conclusion

- Future direction of graph-based explainable recommendation
  - Sentence-based explainable recommendation
    - Providing high quality, fluent sentences for explanation is always a hot topic.
    - Challenge:
      - Bottleneck of natural language processing

4 Conclusion

- Future direction of graph-based explainable recommendation
  - Evaluation for explainable recommendation
    - The evaluation is not sufficient currently. Most of the papers use case study or user study to test the performance of explainability.
    - Challenge:
      - How to define the *good quality* of explanation?
      - How to use *metric* to evaluate?

Thanks for your listening.

3. Graph Contrastive Learning for Recommendation

3.1 Introduction to Contrastive Learning

3.1.1 An illustrative example of contrastive learning

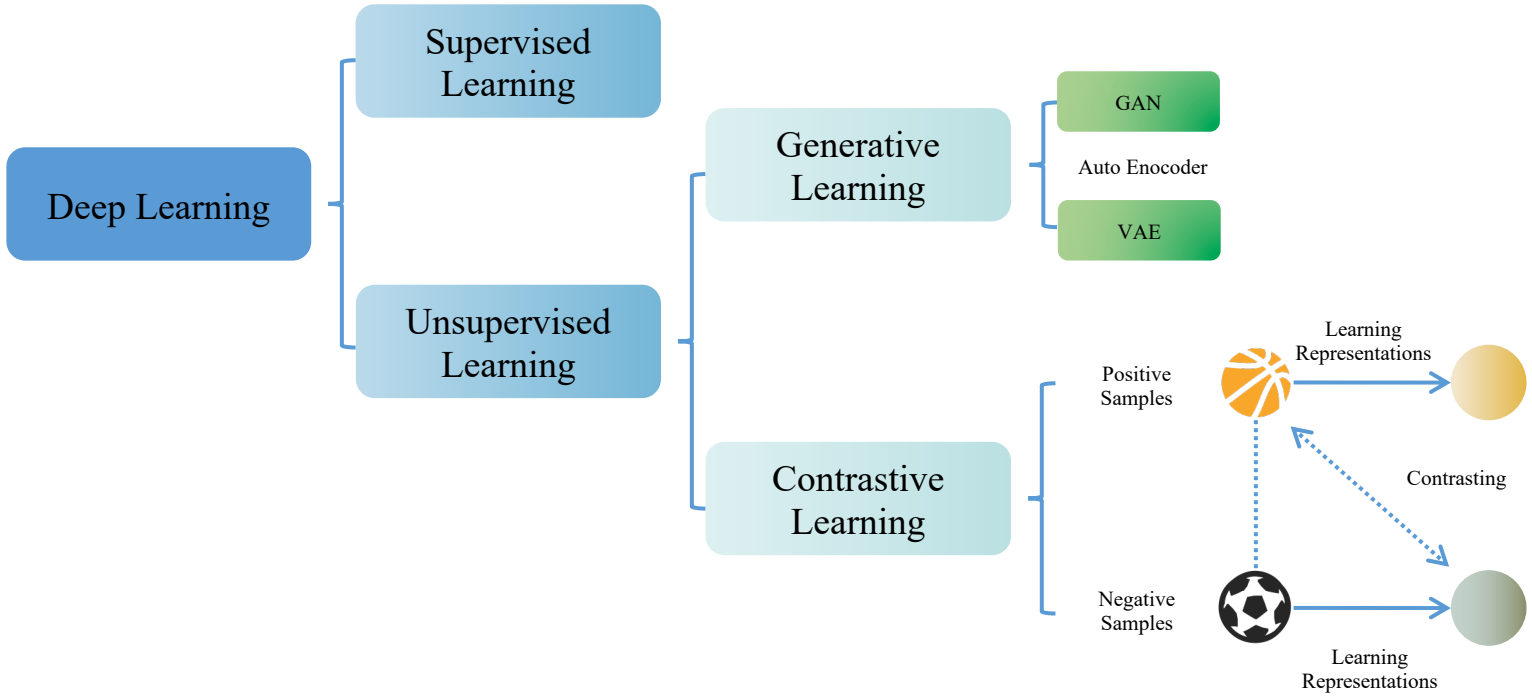
- Limitations of direct-semantic supervision:
  - The underlying data has a much richer structure than what sparse labels or rewards could provide.
  - We cannot rely on direct supervision in high dimensional problems, and the marginal cost of acquiring labels is higher in problems like Reinforcement Learning.
  - It leads to task-specific solutions, rather than knowledge that can be repurposed.

Fig. Left: Drawing of a dollar bill from memory. Right: Drawing subsequently made with a dollar bill present.[1][2]

[1] https://aeon.co/essays/your-brain-does-not-process-information-and-it-is-not-a-computer
[2] https://ankeshanand.com/blog/2020/01/26/contrative-self-supervised-learning.html

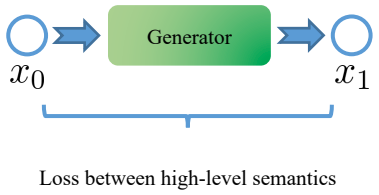
3.1 Introduction to Contrastive Learning

3.1.2 Taxonomy of machine learning schemas

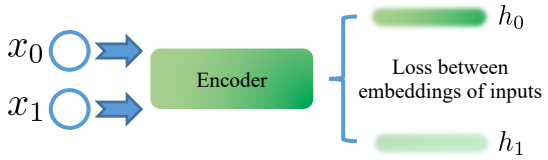


Generative vs Contrastive Learning Methods:

- *Generative or Predictive Methods*
  - Loss measured in the output space



- *Contrastive Methods*
  - Loss measured in the representation space



3.1 Introduction to Contrastive Learning

3.1.3 Contrastive learning objective

- For any data point x, contrastive methods aim to learn an encoder f such that:

$$score(f(x), f(x^+) >> score(f(x), f(x^-))$$

  - here x^+ is the postive sample of x, and x^- is a negative sample.
  - x is commonly referred to as a 'key', and the set of other positive or negative samples are regarded as a 'dictionary'.
  - $score(*)$ is a metric function to measure the similarity of two data points.
  - in practice, researchers usually perform dot product as the score function:

$$score(f(x), f(x^*)) = f(x)^T f(x^*)$$

- InforNCE[1] is one of the widely used loss functions for contrastive learning:

$$\mathcal{L}_{InfoNCE} = -log \frac{e^{f(x)^T f(x^+)}}{\sum_{i=0}^{N} e^{f(x)^T f(x_i)}}$$

  - the set $\{x_0, x_1, \ldots, x_N\}$ contains all the positive and negative samples.
  - this formular is similar to cross-entropy for N-way softmax classification tasks, and commonly called InfoNCE loss in contrastive learning literature.

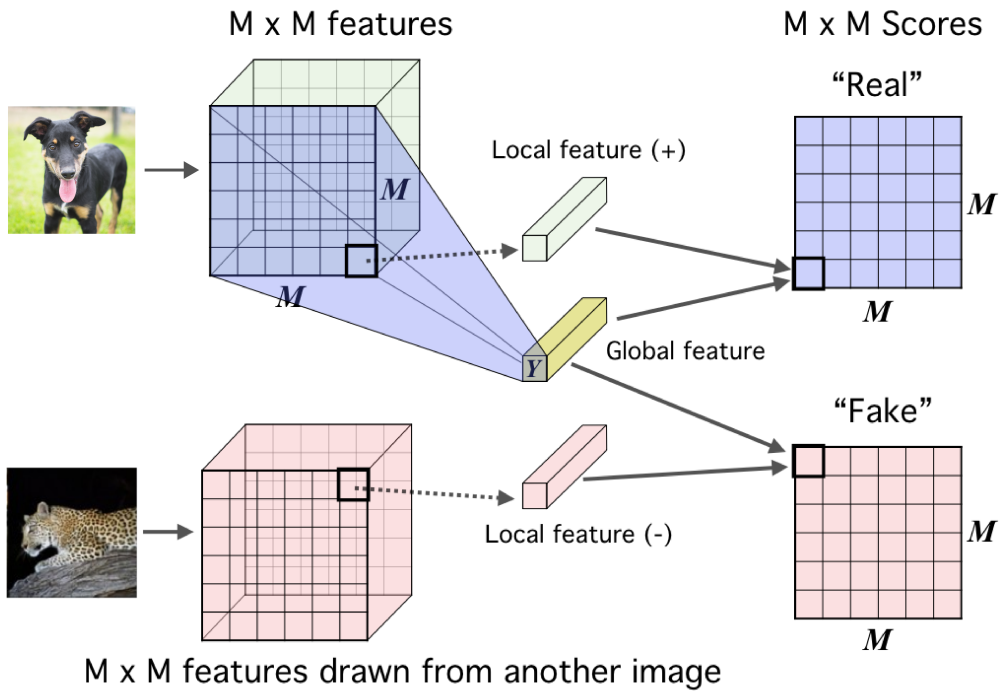


Fig. An example of how to generate 'key' and 'dictionary' to perform contrastive learning from Deep Infomax[2].

[1] Poole, Ben, Sherjil Ozair, Aaron van den Oord, Alexander A. Alemi, and George Tucker. "On variational bounds of mutual information." ICML, 2019.
[2] Hjelm, R. Devon, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. "Learning deep representations by mutual information estimation and maximization." ICLR, 2019

3.2 Introduction to Graph Contrastive Learning

3.2.1 From Deep Infomax to Deep Graph Infomax

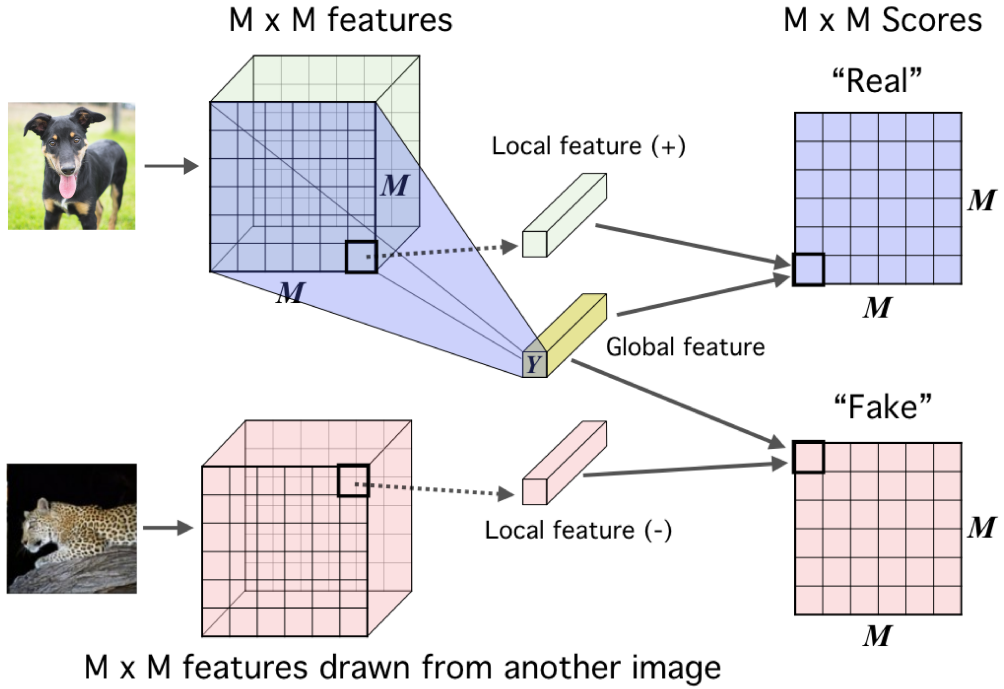


Fig. Overview of Deep Infomax (DIM)[1]

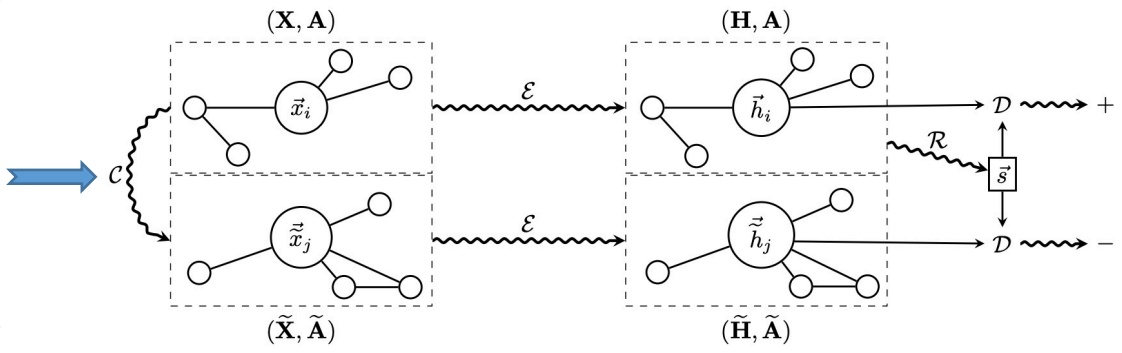
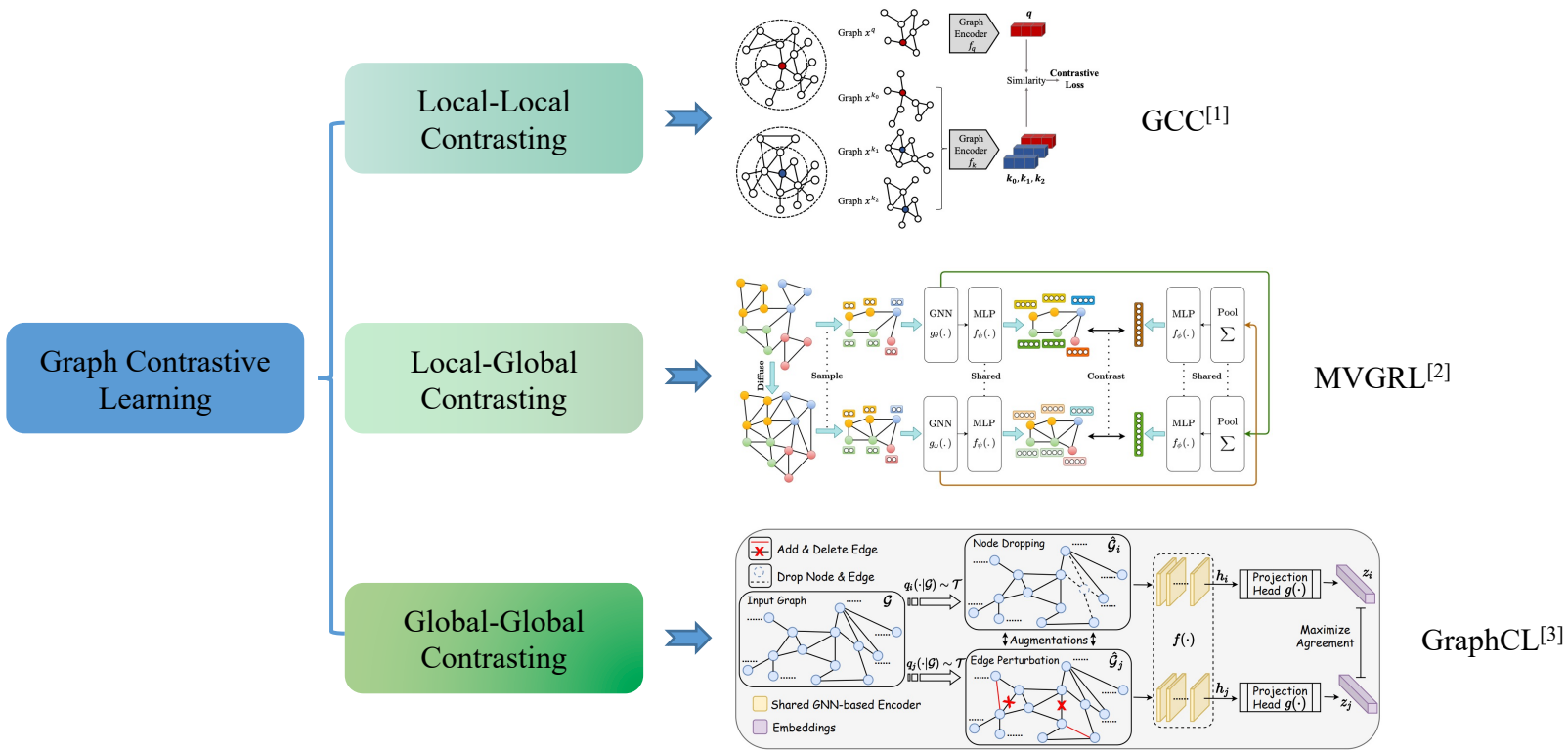


Fig. Overview of Deep Graph Infomax (DGI)[2]

[1] Hjelm, R. Devon, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. "Learning deep representations by mutual information estimation and maximization." ICLR, 2019
[2] Petar Veličković, William Fedus, William L. Hamilton, Pietro Liò, Yoshua Bengio, R Devon Hjelm, "Deep Graph Infomax", ICLR 2019

3.2 Introduction to Graph Contrastive Learning

3.2.2 Taxonomy of Graph Contrastive Learning



- Definitions:
  - *'Local View'* means basic element of the input graph (e.g. edges, nodes) or a small-scale combination of those basic elements (e.g. sub-graphs).
  - *'Global View'* means the whole context of the original input graph or the augmented global views of the input graph.

[1] Jiezhong Qiu，Qibin Chen，Yuxiao Dong，Jing Zhang，Hongxia Yang，Ming Ding，Kuansan Wang，Jie Tang, "GCC: Graph Contrastive Coding for Graph Neural Network Pre-Training", KDD 2020
[2] Kaveh Hassani, Amir Hosein Khasahmadi, "Contrastive Multi-View Representation Learning on Graphs", ICML 2020
[3] Yuning You, Tianlong Chen, Yongduo Sui, Ting Chen, Zhangyang Wang, Yang Shen, "Graph Contrastive Learning with Augmentations", NeurIPS 2020

3.2 Introduction to Graph Contrastive Learning

3.2.3 Graph augmentations for contrasting pairs generation

➢ Graph perturbation:

- *Node dropping* drops nodes and related edges to augment the graph, the underlying prior is that vertex missing does not alter semantics.
- *Edge perturbation* drops or adds some edges in the graph, the underlying prior is that semantic robustness against connectivity variations.
- *Attribute masking* discards partial attributes, the underlying prior is that semantic robustness against losing partial attributes per node.

➢ Sub-graph sampling:

- Sampling a sub-graph from the input graph, the underlying prior is that local structure can hint the full semantics.

Data augmentation	Type	Underlying Prior
Node dropping	Nodes, edges	Vertex missing does not alter semantics.
Edge perturbation	Edges	Semantic robustness against connectivity variations.
Attribute masking	Nodes	Semantic robustness against losing partial attributes per node.
Subgraph	Nodes, edges	Local structure can hint the full semantics.

Tab. Summary of graph augmentation methods.[1]

➢ Usage Scenarios:

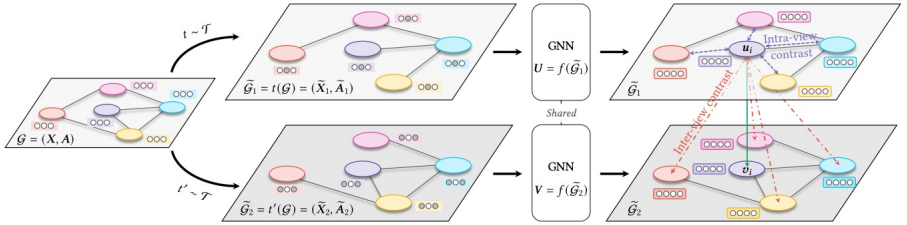
- *Graph perturbation* is usually used in global-global contrasting learning to generate augmented global view.
- *Sub-graph sampling* is usually used in local-local contrasting learning since sub-graph represents local structure of the input graph.

➢ Limitations:

- Graph augmentation schemes, a crucial component in graph contrastive learning, remain rarely explored.
- Most existing methods uniformly augment graph data, like uniformly dropping nodes and edges, and uniformmly shuffling features, leading to suboptimal performances on preserving intrinstic structures and attributes.

➢ Potential Solutions:

- GCA[2] is one of the solutions, which is an augmentation scheme based on node centrality measures to highlight important connective structures:



[1] Yuning You, Tianlong Chen, Yongduo Sui, Ting Chen, Zhangyang Wang, Yang Shen, “Graph Contrastive Learning with Augmentations”, NeurIPS 2020
[2] Yanqiao Zhu, Yichen Xu, Feng Yu, Qiang Liu, Shu Wu, Liang Wang, “Graph Contrastive Learning with Adaptive Augmentations”, WWW 2021

3.2 Introduction to Graph Contrastive Learning

3.2.4 Introduction to *GCC: Graph Contrastive Coding for Graph Neural Network Pre-Training* (1)

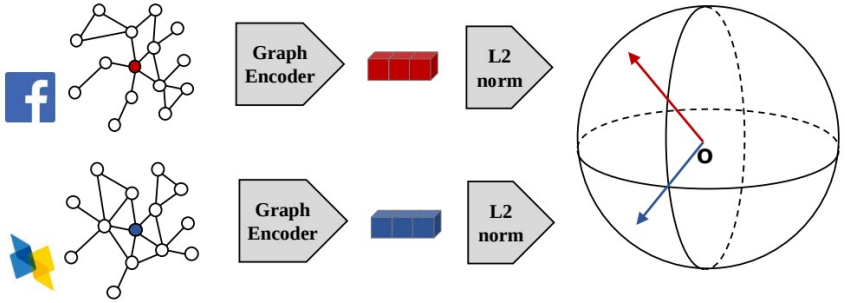


Fig. An illustrative example of GCC[1]

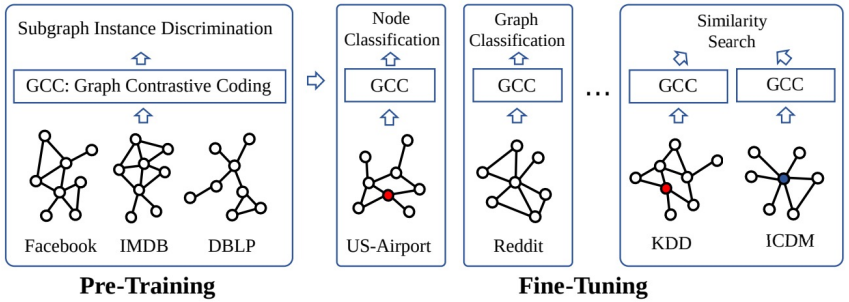


Fig. Training procedures of GCC[1]

- Sub-graph instance discrimination:
  - GCC takes sub-graph instance discrimination as the pre-training task.
  - the goal is to distinguish vertices according to their local structures
  - as the figure shows, the vertex in the Facebook social networks should be different from the vertex in the DBLP citation networks in the embedding space.

- Pre-training and fine-tunning procedures:
  - First, GCC conducts sub-graph instance discrimination tasks among different networks.
  - Then, GCC feeds the pre-trained node embeddings into downstream tasks.
  - Finally, the downstream classifiers or predictors will be fine-tuned.

[1] Jiezhong Qiu，Qibin Chen，Yuxiao Dong，Jing Zhang，Hongxia Yang，Ming Ding，Kuansan Wang，Jie Tang, “GCC: Graph Contrastive Coding for Graph Neural Network Pre-Training”, KDD 2020

3.2 Introduction to Graph Contrastive Learning

3.2.4 Introduction to *GCC: Graph Contrastive Coding for Graph Neural Network Pre-Training* (2)

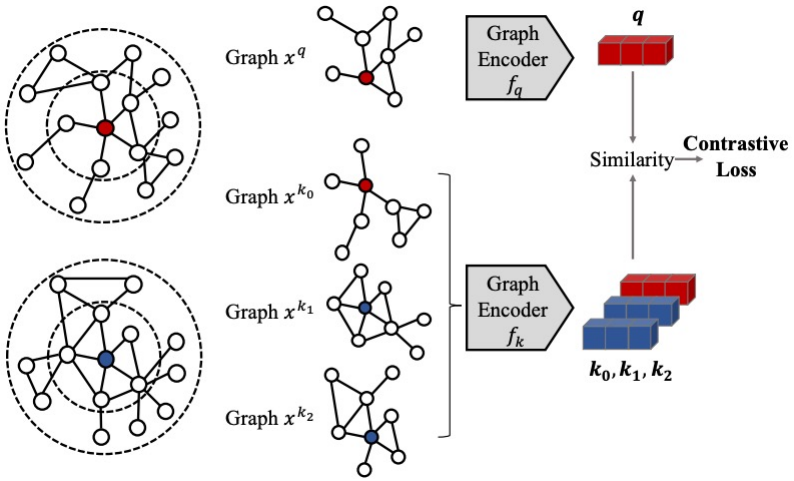


Fig. A running example of GCC[1]

- Instances preparation:
  - GCC first extracts two sub-graphs from the input graph x^q, x^{k_0} as positive samples.
  - then, in this example, the input graph is perturbated to formulate a negative sample, x^{k_1}, x^{k_2} extracted from the negative graph as negative samples.
  - two graph encoders map the sub-graphs to the embeddings, q and $\{k_0, k_1, k_2\}$, and build the bridge between positive and negative samples.

- Training objective:
  - the target of pre-training is to recognize (q, k_0) as a similar pair and distinguish them from negative samples.
  - the loss function can be formulated as:

$$\mathcal{L} = -log \frac{e^{q^T k_0/\tau}}{\sum_{i=0}^{K} e^{q^T k_i/\tau}}$$

[1] Jiezhong Qiu，Qibin Chen，Yuxiao Dong，Jing Zhang，Hongxia Yang，Ming Ding，Kuansan Wang，Jie Tang,“GCC: Graph Contrastive Coding for Graph Neural Network Pre-Training”, KDD 2020

3.2 Introduction to Graph Contrastive Learning

3.2.5 Introduction to *Contrastive Multi-View Representation Learning on Graphs* (1)

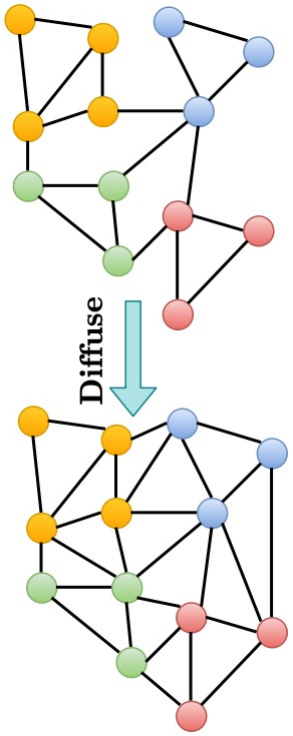


Fig. Graph diffusion procedures of MVGRL.[3]

➢ Graph diffussion:

- Graph diffusion is formulated as the following equation[1], where $T \in R^{n\times n}$ is the generalized transition matrix and Θ is the weighting coefficient which determines the ratio of global-local information:

$$\mathbf{S} = \sum_{k=0}^{\infty} \Theta_k \mathbf{T}^k \in \mathbb{R}^{n\times n}$$

➢ Graph diffussion methods:

- *Personalized PageRank*[2] *(PPR)* Given an adjacency matrix $A \in R^{n\times n}$, a diagonal degree matrix $D \in R^{n\times n}$, and α denotes teleport probaility in a random walk, PPR can be formulated as:

$$\mathbf{S}^{PPR} = \alpha(\mathbf{I}_n - (1-\alpha)\mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2})^{-1}$$

- *Heat Kernel*[1] Given an adjacency matrix $A \in R^{n\times n}$, a diagonal degree matrix $D \in R^{n\times n}$, and t denotes the diffusion time, heat kernel method can be formulated as:

$$\mathbf{S}^{heat} = \exp(t\mathbf{A}\mathbf{D}^{-1} - t)$$

[1] Johannes Klicpera, Stefan Weißenberger, Stephan Günnemann, “Diffusion Improves Graph Learning”, NeurIPS 2019
[2] Page, Lawrence, Brin, Sergey, Motwani, Rajeev and Winograd, Terry The PageRank Citation Ranking: Bringing Order to the Web. , Stanford Digital Library Technologies Project (1998)
[3] Kaveh Hassani, Amir Hosein Khasahmadi, “Contrastive Multi-View Representation Learning on Graphs”, ICML 2020

3.2 Introduction to Graph Contrastive Learning

3.2.6 Introduction to *Contrastive Multi-View Representation Learning on Graphs* (2)

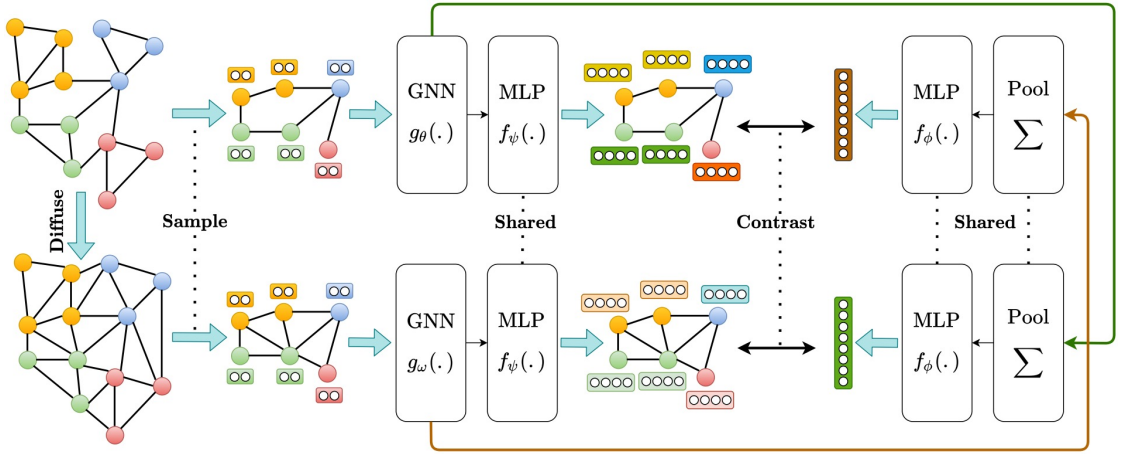


Fig. Overview of MVGCL[1]

- Training Objective:
  - Let $\theta, \omega, \phi, \psi$ denote graph encoder and projection head parameters, respectively.
  - Given $\vec{h}_i^{\,\alpha}, \vec{h}_j^{\,\beta}$ denote the representations of node i and graph g encoded from two different views, respectively.
  - The training objective is formulated as:

$$\max_{\theta,\omega,\phi,\psi} \frac{1}{|\mathcal{G}|} \sum_{g \in \mathcal{G}} \left[\frac{1}{g} \sum_{i=1}^{|g|} \left[\mathbf{MI}(\vec{h}_i^{\,\alpha}, \vec{h}_g^{\,\beta}) + \mathbf{MI}(\vec{h}_i^{\,\beta}, \vec{h}_g^{\,\alpha}) \right] \right]$$

[1] Kaveh Hassani, Amir Hosein Khasahmadi, “Contrastive Multi-View Representation Learning on Graphs”, ICML 2020

3.2 Introduction to Graph Contrastive Learning

3.2.7 Introduction to *Graph Contrastive Learning with Augmentations*

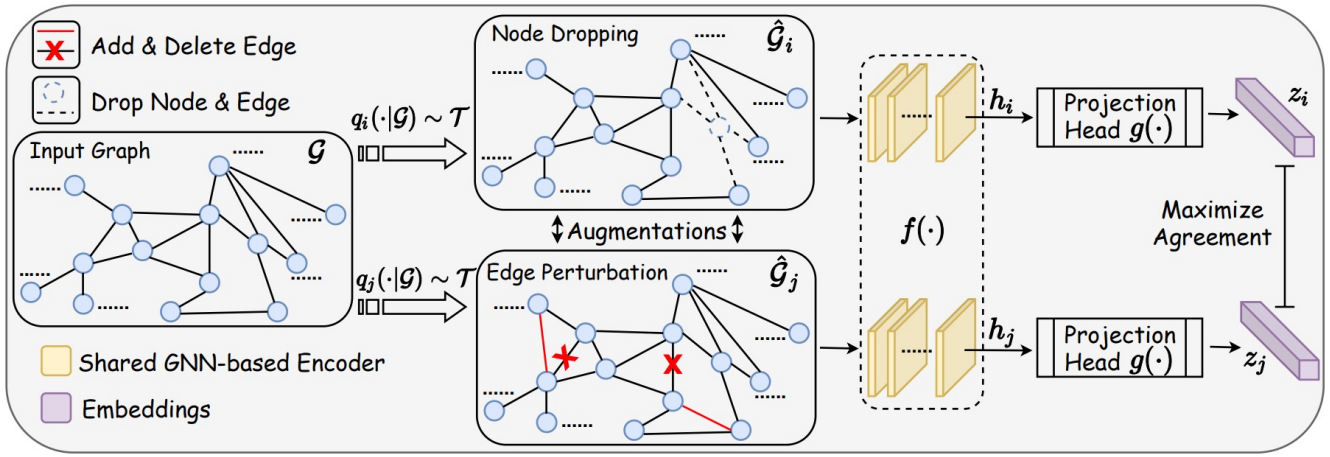


Fig. Overview of GraphCL[1]

- Augmented graphs generation:
  - GraphCL take different graph perturbation methods, including node dropping and edge perturbation to generate the augmented graphs.
  - assume that there N graphs in a minibatch, resulting in $2N$ augmented graphs in total.
  - let $z_{n,i}, z_{n,j}$ denote embeddings of the two augmented graphs of the n-th graph in the minibatch.

- Pre-training procedures and objectives:
  - GraphCL claims that minor perturbations will not change the semantics, so one of the training objective is to maximize the similarity between two augmented graphs.
  - note that we need negative samples to fulfill the contrastive learning procedures, GraphCL does not explicitly sample negative pairs, but utilize other graphs in the same minbatch.
  - The loss function can be formulated as:

$$\mathcal{L} = -log\frac{exp(sim(z_{n,i}, z_{n,j})/\tau)}{\sum_{n'=1,n'\neq n}^{N} exp(sim(z_{n,i}, z_{n',j})/\tau)}$$

[1] Yuning You, Tianlong Chen, Yongduo Sui, Ting Chen, Zhangyang Wang, Yang Shen, “Graph Contrastive Learning with Augmentations”, NeurIPS 2020

3.3 Graph Contrastive Learning for Recommendation

3.3.1 Background

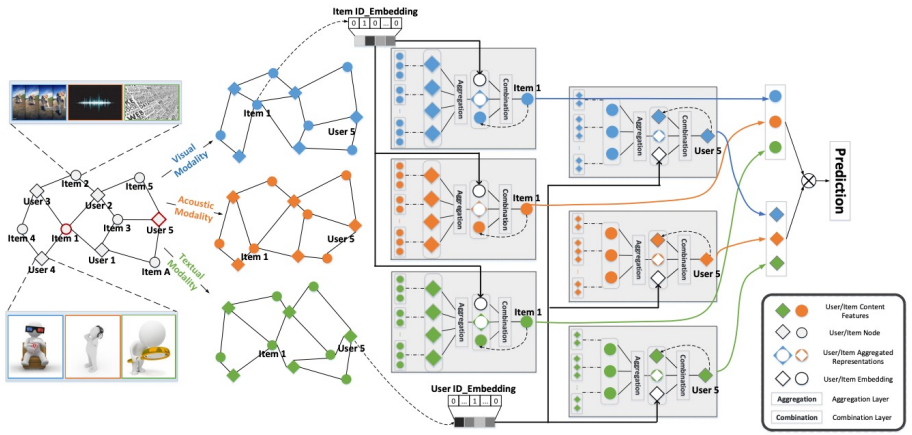


Fig. An Example of Graph Learning for Recommendation[1]

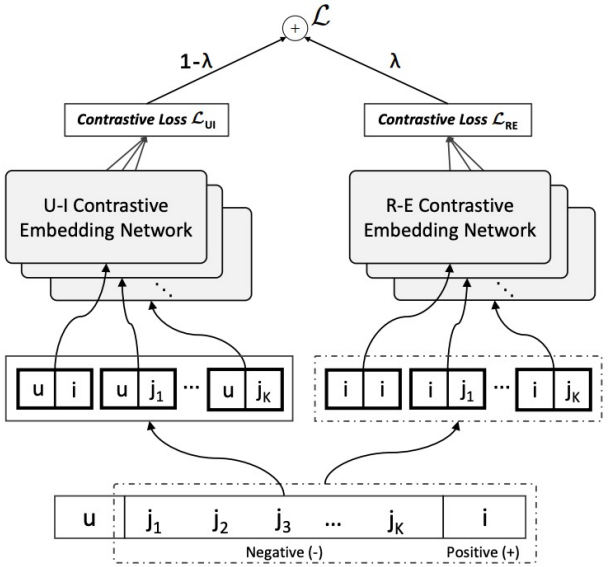


Fig. An Example of Contrastive Learning for Recommendation[2]

Can we combine these two advanced techniques to formulate graph contrastive learning methods for recommendation?

[1] Yinwei Wei, Xiang Wang, Liqiang Nie, Xiangnan He, Richang Hong, and Tat-Seng Chua, “MMGCN: Multi-modal Graph Convolution Network for Personalized Recommendation of Micro-video”, ACM Multimedia 2019
[2] Yinwei Wei, Xiang Wang, Qi Li, Liqiang Nie, Yan Li, Xuanping Li, Tat-Seng Chua, “Contrastive Learning for Cold-Start Recommendation”, ACM Multimedia 2021

3.3 Graph Contrastive Learning for Recommendation

3.3.2 Introduction to *Hyper Meta-Path Contrastive Learning for Multi-Behavior Recommendation* (1)

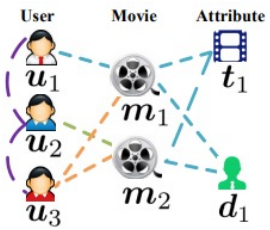
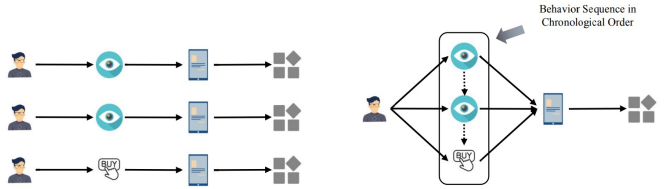


Fig. Heterogeneous information network for movie recommendation.[1]



(a) Meta-paths of a user (b) A hyper meta-path of a user

Fig. (a) Meta-paths of a user in the recommendation system denotes the user's different behaviors. (b) A hyper meta-path constructed by previous meta-paths denotes the user's behavior pattern when he is purchasing a smartphone[2]

So many schemas, which should we take?

$$U \rightarrow M \rightarrow T$$
$$U \rightarrow M \rightarrow D$$
$$U \rightarrow M \rightarrow U$$
$$U \rightarrow M \rightarrow T \rightarrow U$$

- Hyper Meta-Path: A *hyper meta-path* is a logical composition of multiple meta-path schemas connecting two end nodes in a heterogeneous information network. Hyper meta-path has the following properties:
  - It describes the logical relations (e.g., chronological order, spatial order and topological order) among a sort of of meta-paths with the same end nodes.
  - Multiple hyper meta-paths, which have the same start node, compose a *hyper meta-graph*.

[1] Binbin Hu, Chuan Shi, Wayne Xin Zhao and Philip S. Yu, “Leverage Meta-path based Context for Top-N Recommendation with a Neural Co-Attention Model”, KDD 2018
[2] Haoran Yang, Hongxu Chen, Lin Li, Philip S. Yu, and Guandong Xu, “Hyper Meta-Path Contrastive Learning for Multi-Behavior Recommendation”, ICDM 2021

3.3 Graph Contrastive Learning for Recommendation

3.3.3 Introduction to *Hyper Meta-Path Contrastive Learning for Multi-Behavior Recommendation* (2)

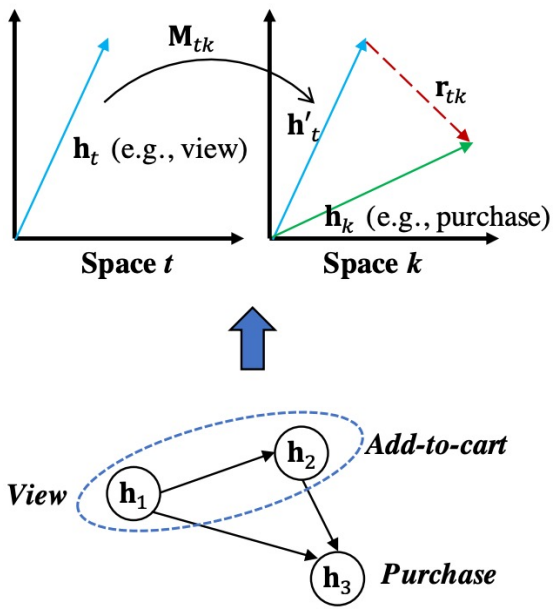


Fig. Behavior embeddings transformation in EHCF.[1]

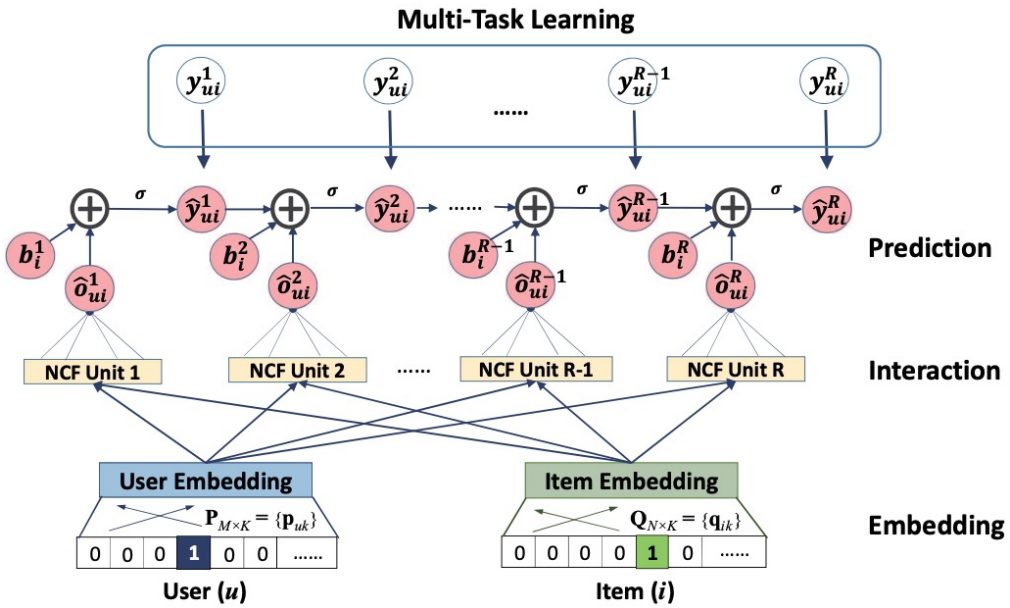


Fig. Relations among different behaviors in NMTR[2]

Fixed learning schema is hard to capture the complex dependencies among different users' behaviors.

[1] C. Chen, Min Zhang, Yongfeng Zhang, Weizhi Ma, Yiqun Liu, Shaoping Ma, "Efficient Heterogeneous Collaborative Filtering without Negative Sampling for Recommendation", AAAI 2020
[2] Chen Gao, Xiangnan He, Dahua Gan, Xiangning Chen, Fuli Feng, Yong Li, Tat-Seng Chua, Lina Yao, Yang Song, Depeng Jin, "Learning to Recommend with Multiple Cascading Behaviors", TKDE

3.3 Graph Contrastive Learning for Recommendation

3.3.4 Introduction to *Hyper Meta-Path Contrastive Learning for Multi-Behavior Recommendation* (3)

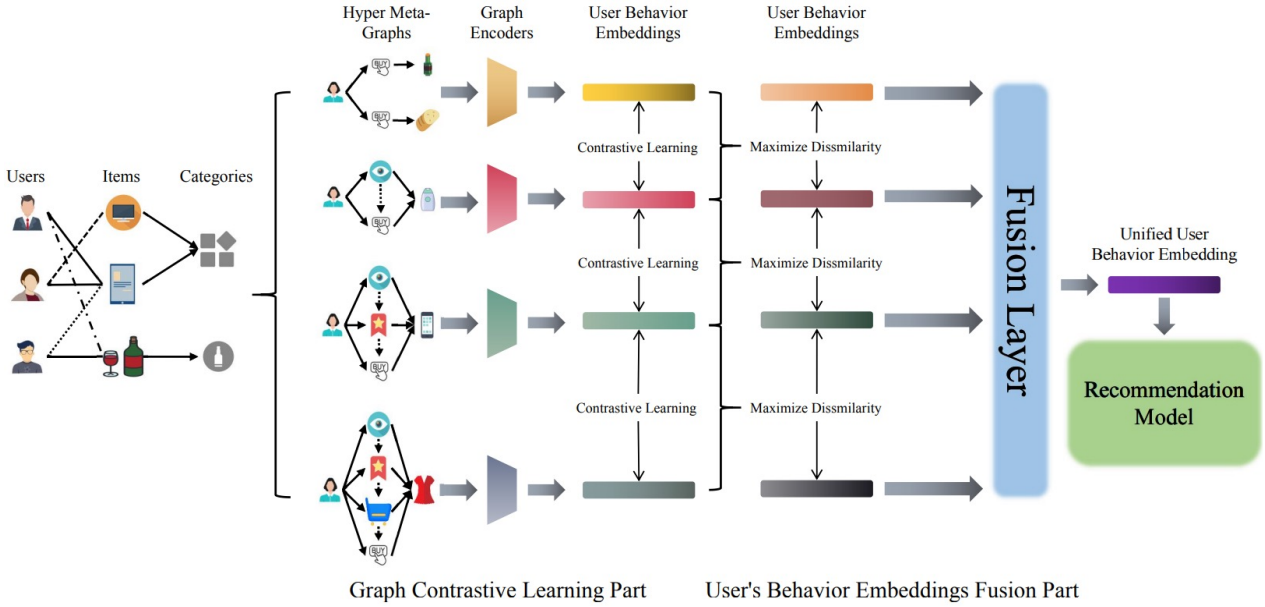


Fig. Overview of HMG-CR.[1]

- ➢ Hyper meta-graph generation:
  - Construct hyper meta-paths according to the choronological order among users' beahviors to formulate hyper meta-graph.
  - Add different types of behaviors into the graph incrementally.

- ➢ Contrastive learning among hyper meta-graphs:
  - different hyper meta-graphs contain different types of users' behaviors
  - contrasting among different graphs can obtain users' different purchasing patterns towards different products.

[1] Haoran Yang, Hongxu Chen, Lin Li, Philip S. Yu, and Guandong Xu, “Hyper Meta-Path Contrastive Learning for Multi-Behavior Recommendation”, ICDM 2021

3.3 Graph Contrastive Learning for Recommendation

3.3.5 Introduction to *Hyper Meta-Path Contrastive Learning for Multi-Behavior Recommendation* (4)

Dataset	Taobao				Tmall			
Methods \ Metrics	Recall@5	Recall@10	NDCG@5	NDCG@10	Recall@5	Recall@10	NDCG@5	NDCG@10
GCN	0.2577	0.3589	0.1842	0.2167	0.2544	0.3775	0.1763	0.2163
GraphSAGE	0.2751	0.3826	0.1965	0.2312	0.2588	0.3695	0.1813	0.2170
GAT	0.2782	0.3921	0.1972	0.2339	0.2561	0.3735	0.1777	0.2158
RGCN	0.2714	0.3767	0.1946	0.2285	0.2725	0.4144	0.1749	0.2215
NMTR	0.2215	0.3781	0.1513	0.2012	0.2780	0.4230	0.1798	0.2265
EHCF	0.2882	0.4166	0.1945	0.2359	0.2451	0.4115	0.1581	0.2113
HMG-CR(SG)	0.3050	0.4417	0.2162	0.2608	0.2943	0.4329	0.1863	0.2321
HMG-CR(GCN)	0.3039	0.4441	0.2154	0.2613	0.2954	0.4332	0.1869	0.2324
HMG-CR(GAT)	0.3460	0.4390	0.2443	0.2746	0.3163	0.4320	0.2224	0.2604
HMG-CR(GIN)	0.3141	0.3627	0.2029	0.2191	**0.3547**	0.4313	**0.2642**	**0.2891**
HMG-CR(TAG)	**0.3588**	**0.4464**	**0.2639**	**0.2926**	0.2964	**0.4350**	0.1902	0.2359
Improvement	24.50%	7.15%	33.82%	24.04%	27.59%	2.84%	45.73%	27.64%

Tab. Comparison experiment results of HMG-CR.[1]

- Comparison experiment results analysis:
  - HMG-CR is a flexible framework which can be coupled with multiple graph neural network models
  - Graph topology or structure -awared GNNs, including GIN and TAG are more suitable to conduct graph contrastive learning
  - HMG-CR has better performances on NDCG.

[1] Haoran Yang, Hongxu Chen, Lin Li, Philip S. Yu, and Guandong Xu, “Hyper Meta-Path Contrastive Learning for Multi-Behavior Recommendation”, ICDM 2021

3.3 Graph Contrastive Learning for Recommendation

3.3.6 Introduction to HCGR: Hyperbolic Contrastive Graph Representation Learning for Session-based Recommendation (1)

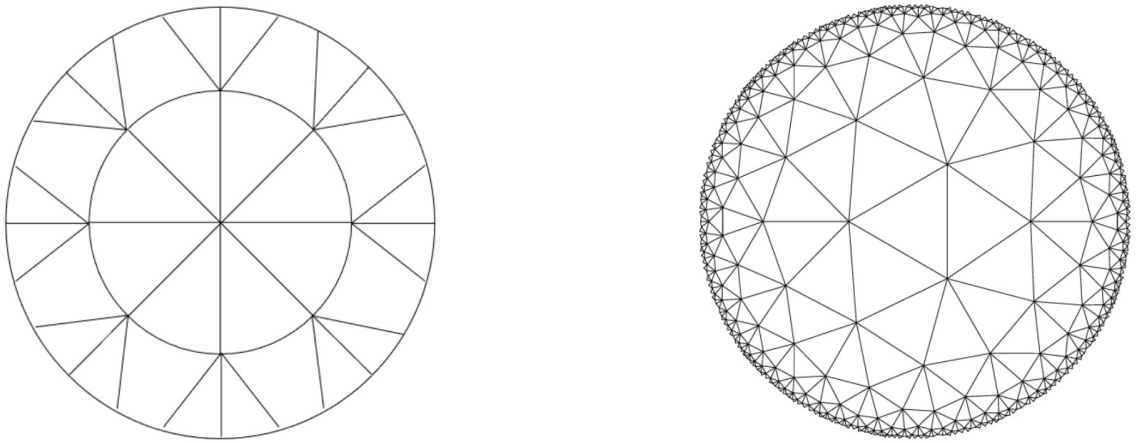


Fig. Left: Euclidean space. Right: 2D Poincare disk.[1]

- Challenge 1:

  The user's interests are extensive and hierarchical, which can be expressed as a power-law distribution of items clicked by users. The existing session-based recommendation methods learn the representations in Euclidean space, which can't effectively capture the information of such hierarchical, or in other words, tree-structured data.

- Challenge 2:
  - Recent studies have proved that hierarchical data can be better explained under Non-Euclidean geometry of low-dimensional manifolds. But in the GNN based methods, introducing Non-Euclidean transformation will result in the discrepancy between Euclidean and Non-Euclidean space when aggregating neighbor messages and applying attention mechanism.

[1] Naicheng Guo, Xiaolei Liu, Shaoshuai Li, Qiongxu Ma, Yunan Zhao, Bing Han, Lin Zheng, Kaixin Gao, Xiaobo Guo, "HCGR: Hyperbolic Contrastive Graph Representation Learning for Session-based Recommendation", arXiv 2021

3.3 Graph Contrastive Learning for Recommendation

3.3.7 Introduction to HCGR: Hyperbolic Contrastive Graph Representation Learning for Session-based Recommendation (2)

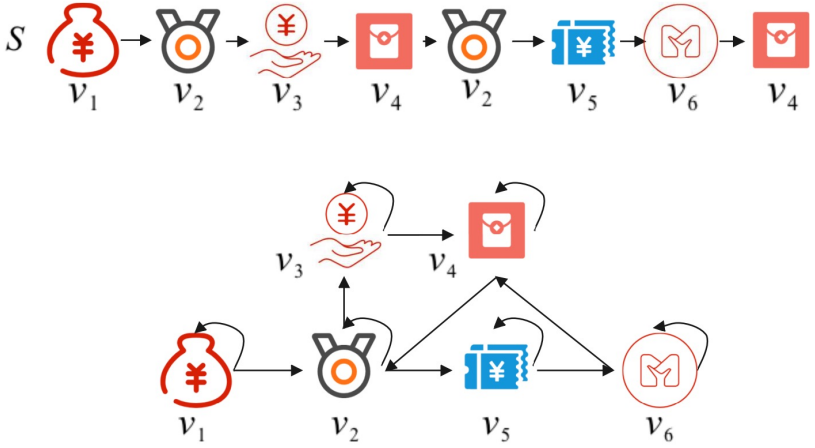


Fig. An example of how to convert a session to graph.[1]

- Session graph:
  - Graph neural network models cannot deal with session directly unless converting the session into the session graph.
  - Users' behaviors are choronological and items are usually get involved in multiple transactions, directed graph is a good choice to demonstrate the session.

[1] Naicheng Guo, Xiaolei Liu, Shaoshuai Li, Qiongxu Ma, Yunan Zhao, Bing Han, Lin Zheng, Kaixin Gao, Xiaobo Guo, “HCGR: Hyperbolic Contrastive Graph Representation Learning for Session-based Recommendation”, arXiv 2021

3.3 Graph Contrastive Learning for Recommendation

3.3.8 Introduction to HCGR: Hyperbolic Contrastive Graph Representation Learning for Session-based Recommendation (3)

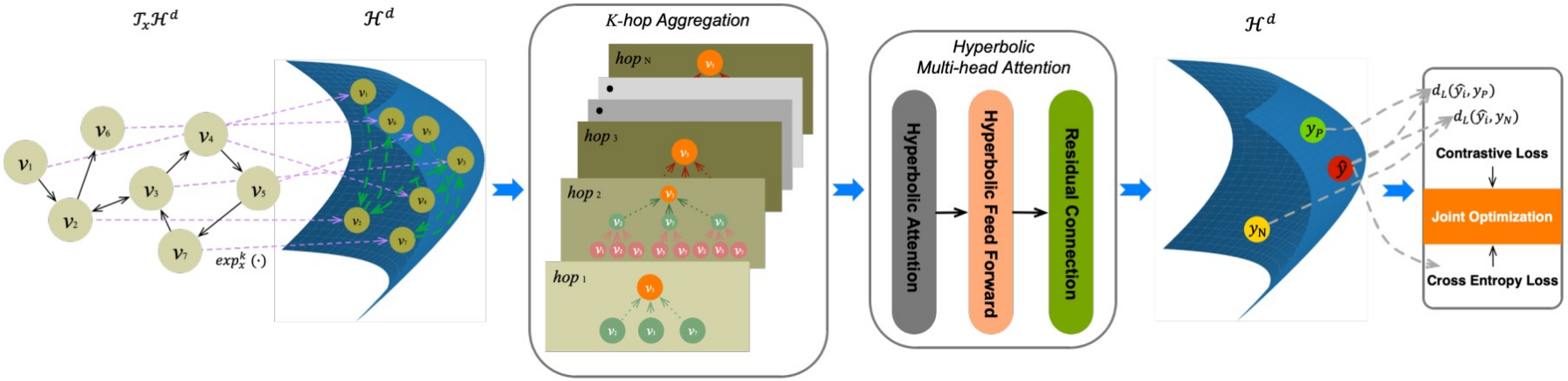


Fig. Overview of HCGR.[1]

- Methodology overview:
  - *Space mapping* maps Euclidean features to Hyperbolic features.
  - *K-hop Aggregation* conduct aggregation and pooling among neighbor nodes in the graph.
  - *Hyperbolic Multi-head Attention* introduces attention mechanism in the Hyperbolic space to enhance the performances.
  - *Training Objectives* contain two loss functions, inclduing contrasstive loss and cross-entropy loss.

[1] Naicheng Guo, Xiaolei Liu, Shaoshuai Li, Qiongxu Ma, Yunan Zhao, Bing Han, Lin Zheng, Kaixin Gao, Xiaobo Guo, “HCGR: Hyperbolic Contrastive Graph Representation Learning for Session-based Recommendation”, arXiv 2021

3.3 Graph Contrastive Learning for Recommendation

3.3.9 Introduction to HCGR: Hyperbolic Contrastive Graph Representation Learning for Session-based Recommendation (4)

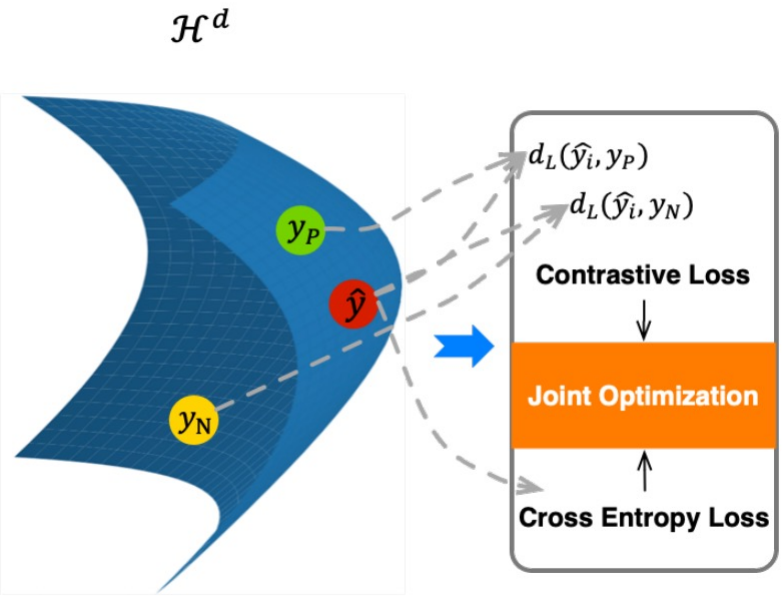


Fig. Training objectives of HCGR.[1]

- Contrastive loss:
  - Too many similar items in the recommendation systems, but users just choose thier favorite ones.
  - If the model can distinguish the subtle distinction, it may significantly improve the ranking performances.
  - Inspired by the success of contrastive learning, HCGR utilizes it to separate the positive and negative pairs up to given margin:

$$\mathcal{L}_c = \sum_{i=1}^{n} \max(d_L(\hat{y}_i, y_P) - d_L(\hat{y}_i, y_N) + \xi, 0)$$

- Cross-entropy loss:
  - Unsupervised signals iare not enough, HCGR introduces supervised signals to enhance the model:

$$\mathcal{L}_e = -\sum_{i=1}^{n} (y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i))$$

[1] Naicheng Guo, Xiaolei Liu, Shaoshuai Li, Qiongxu Ma, Yunan Zhao, Bing Han, Lin Zheng, Kaixin Gao, Xiaobo Guo, “HCGR: Hyperbolic Contrastive Graph Representation Learning for Session-based Recommendation”, arXiv 2021

3.3 Graph Contrastive Learning for Recommendation

3.3.10 Introduction to HCGR: Hyperbolic Contrastive Graph Representation Learning for Session-based Recommendation (5)

Dataset	Metric	FPMC	FOSSIL	GRU4Rec	NARM	SASRec	STAMP	SRGNN	GC-SAN	FGNN	LESSR	**HCGR**	**Improv.**
Last.FM	H@10	0.0623	0.0639	0.0759	0.0749	0.0808	0.0735	0.0902	0.0939	0.0946	0.106	**0.1071**	**1.04%**
	M@10	0.03	0.0178	0.0288	0.0276	0.0342	0.0343	0.0411	0.0419	0.0414	0.0457	**0.0523**	**14.44%**
	N@10	0.0376	0.0286	0.0398	0.0386	0.0452	0.0411	0.0484	0.0542	0.0539	0.062	**0.0651**	**5.00%**
	H@20	0.0923	0.0951	0.1131	0.1153	0.1192	0.1177	0.1099	0.1208	0.1241	0.1275	**0.1388**	**8.86%**
	M@20	0.0321	0.02	0.0312	0.0303	0.037	0.0362	0.0431	0.0437	0.0434	0.0505	**0.0541**	**7.13%**
	N@20	0.0452	0.0365	0.0491	0.0488	0.0549	0.0378	0.0354	0.0611	0.0614	0.0699	**0.0749**	**7.15%**
Yoochoose	H@10	0.4093	0.4014	0.4524	0.4615	0.4317	0.3967	0.4341	0.4768	0.4642	0.4735	**0.4798**	**0.61%**
	M@10	0.1603	0.1471	0.2163	0.2207	0.1716	0.1915	0.2204	0.188	0.197	0.2241	**0.2253**	**0.54%**
	N@10	0.219	0.2072	0.2719	0.2773	0.2328	0.2401	0.2709	0.2558	0.2598	0.2828	**0.2898**	**2.48%**
	H@20	0.5013	0.4902	0.5544	0.5636	0.5391	0.4797	0.5279	0.5895	0.5687	0.5722	**0.5938**	**0.73%**
	M@20	0.1668	0.1533	0.2235	0.2278	0.1791	0.1973	0.227	0.1959	0.2044	0.231	**0.2325**	**0.65%**
	N@20	0.2424	0.2297	0.2978	0.3032	0.26	0.2611	0.2946	0.2844	0.2864	0.3078	**0.3162**	**2.73%**
Ta-Feng	H@10	0.0853	0.0995	0.1091	0.1028	0.1091	0.0861	0.094	0.1099	0.1056	0.1115	**0.1134**	**1.70%**
	M@10	0.04	0.0344	0.0456	0.0438	0.0447	0.0404	0.0435	0.0444	0.0396	0.0378	**0.0487**	**28.84%**
	N@10	0.0506	0.0497	0.0604	0.0576	0.0598	0.0511	0.0554	0.0587	0.0552	0.0533	**0.0539**	**1.13%**
	H@20	0.1149	0.1358	0.1509	0.1401	0.1494	0.1181	0.1262	0.1403	0.1424	0.1477	**0.1507**	**2.03%**
	M@20	0.042	0.0369	0.0485	0.0464	0.0475	0.0426	0.0458	0.0472	0.0422	0.0489	**0.0512**	**4.70%**
	N@20	0.058	0.0589	0.0709	0.067	0.07	0.0592	0.0635	0.0699	0.0644	0.0673	**0.0733**	**8.92%**
MYbank	H@10	0.5136	0.4521	0.5647	0.5459	0.5232	0.5542	0.5522	0.5505	0.5612	0.5562	**0.5773**	**2.21%**
	M@10	0.2899	0.2623	0.3255	0.3185	0.3016	0.3167	0.3173	0.3164	0.3299	0.3104	**0.3373**	**2.24%**
	N@10	0.3429	0.3073	0.3822	0.3724	0.354	0.373	0.373	0.3754	0.3835	0.3811	**0.3901**	**1.72%**
	H@20	0.6185	0.5438	0.6647	0.6453	0.6261	0.6603	0.6544	0.6581	0.6684	0.6616	**0.6713**	**0.43%**
	M@20	0.2972	0.2686	0.3324	0.3254	0.3087	0.3241	0.3245	0.3417	0.3445	0.3424	**0.3578**	**3.86%**
	N@20	0.3694	0.3304	0.4075	0.3976	0.3799	0.3998	0.3989	0.4025	0.4103	0.4026	**0.4135**	**0.78%**

* Realtive improvemens are calculated by comparing with the second best performance

Tab. Comparison experiment results of HCGR.[1]

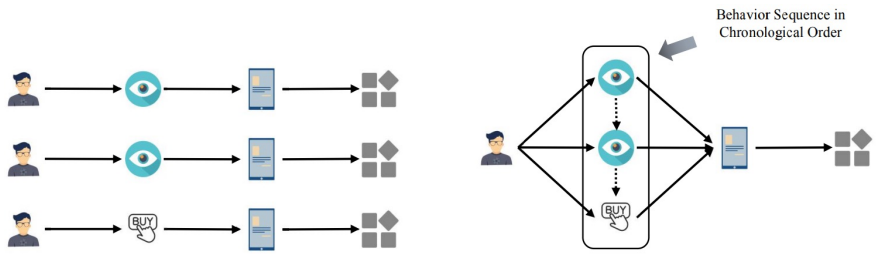
- Finding 1:
  The GNN-based models achieve better performance than RNNs-based models with or without attention mechanism due to the remarkable capacity of graph neural networks to capture complex interaction of user behaviors and describe the coherence of items in a session, which are ignored by RNNs-based models and such ignorance leads to overfitting in RNNs-based models.

- Finding 2:
  HCGR consistently outperforms all the comparison models on all datasets. Compared with FGNN and LESSR, the model involves an advanced hyperbolic learning component to more effectively capture the coherence and hierarchy representations of the user behaviors within the Lorentz hyperbolic space, which ensures the correctness of the necessary representations' transformation.

[1] Naicheng Guo, Xiaolei Liu, Shaoshuai Li, Qiongxu Ma, Yunan Zhao, Bing Han, Lin Zheng, Kaixin Gao, Xiaobo Guo, "HCGR: Hyperbolic Contrastive Graph Representation Learning for Session-based Recommendation", arXiv 2021

3.4 Summary and Future Work of Graph Contrastive Learning for Recommendation

3.4.1 Summary



(a) Meta-paths of a user (b) A hyper meta-path of a user

Fig. Construct proper graphs to prepare to generate contrasting pairs.[1]

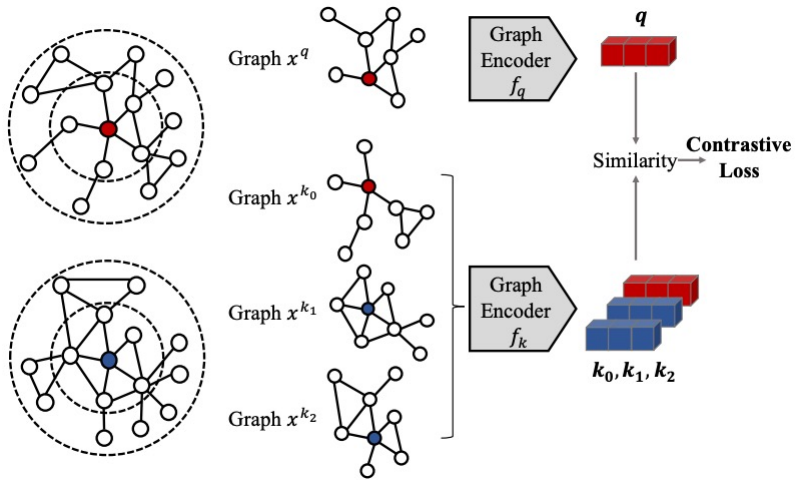


Fig. Formulate proper contrasting procedures.[2]

- Graph generation:
  - With no graph data, we cannot utilize graph neural network models.
  - With no proper and well-orgnized graph, we cannot construct promising contrasting pairs to conduct graph contrastive learning.
- Formulating contrasting procedures:
  - Promising contrasting procedures can help us to leverage the advantages of graph contrastive learning.
  - For example, HMG-CR's strategy is contrasting among graphs containing different user behaviors to obtain the hidden relations between these behaviors.
  - However, HGCR, though its name is “Hyperbolic Contrastive Graph Representation Learning”, the contrastive learning is conducted on semantic level instead of graph structure level.

[1] Haoran Yang, Hongxu Chen, Lin Li, Philip S. Yu, and Guandong Xu, “Hyper Meta-Path Contrastive Learning for Multi-Behavior Recommendation”, ICDM 2021
[2] Jiezhong Qiu，Qibin Chen，Yuxiao Dong，Jing Zhang，Hongxia Yang，Ming Ding，Kuansan Wang，Jie Tang, “GCC: Graph Contrastive Coding for Graph Neural Network Pre-Training”, KDD 2020

3.4 Summary and Future Work of Graph Contrastive Learning for Recommendation

3.4.1 Future work

- Constrasting pair generation
  - How to utilize existing graph augmentation methods to generate proper graph instances in recommendation systems ?
  - How to make sure the generate negative samples for contrastive learning is hard negative samples?

- Contrasting procedures formulation
  - In real-world problems, e.g., recommendation systems, graph data has much more informative semantics, we cannot solely focus on the graph structure contrastive learning.

Works about graph contrastive learning for recommendation is not fully explored, and the related literatures are scarce. With the success of graph contrastive learning, we believe it will be a novel and useful tool for graph based recommendation systems being waiting for researchers' explorations.

Thanks for your listening.