# Heterogeneous Graph Neural Network: Models and Applications

Houye Ji

DMGroup

Beijing University of Posts and Telecommunications

2020.01.16

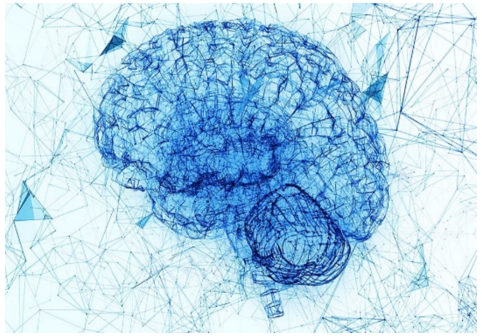
CONTENTS

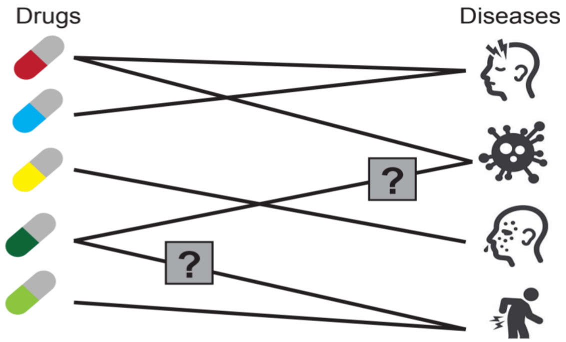
# CONTENTS

# Graph-structured Data and Graph Mining

## Graph-structured Data

- Graph-structured data are ubiquitous.
- Graph-structured data are flexible to model complex interactions.



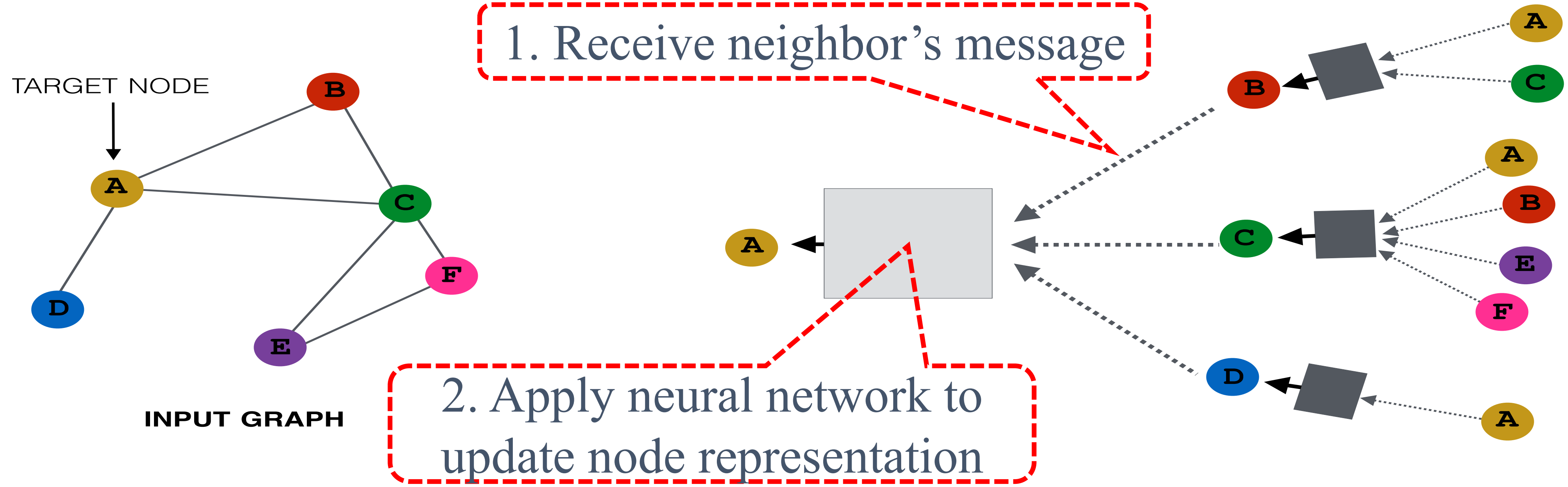

## Graph Mining

- Graph mining is an important component of data mining.
- Graph mining creates tremendous economic value in industry(e.g., PageRank).

## Graph Neural Network

- Neural networks for processing graph-structured inputs.
- Flexible to characterize non-Euclidean data.
- Reasoning and inference.

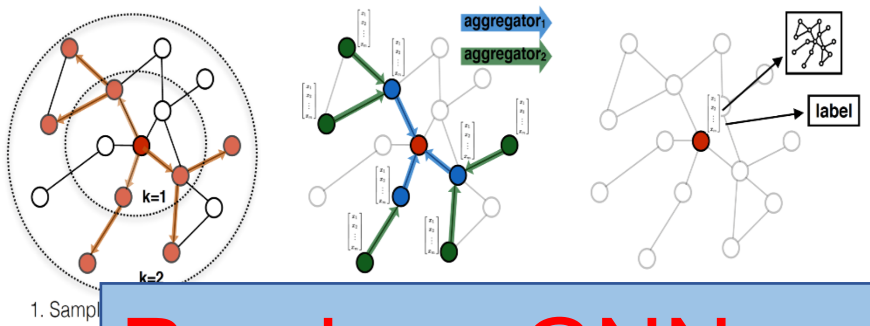
## Two steps of GNN

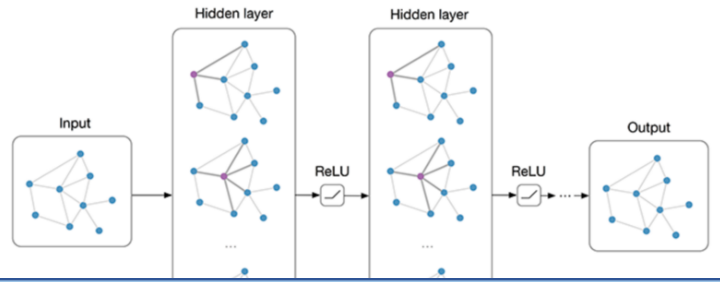


## Architectures of Graph Neural Networks

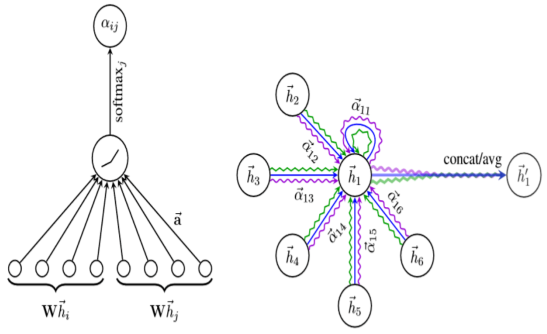
GraphSAGE

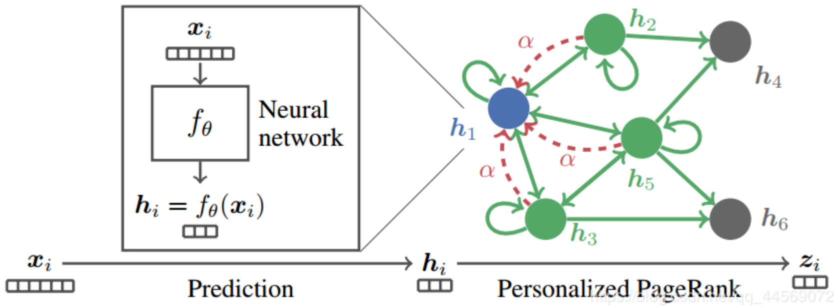
GCN



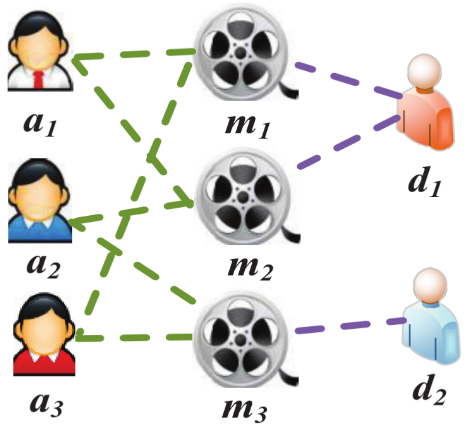

Previous GNNs cannot be directly applied to heterogeneous graph!

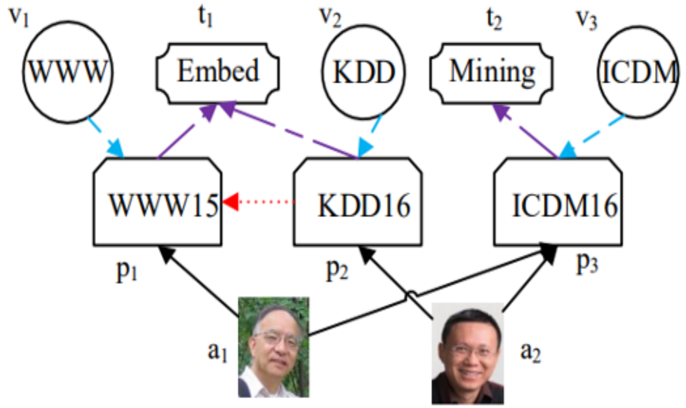


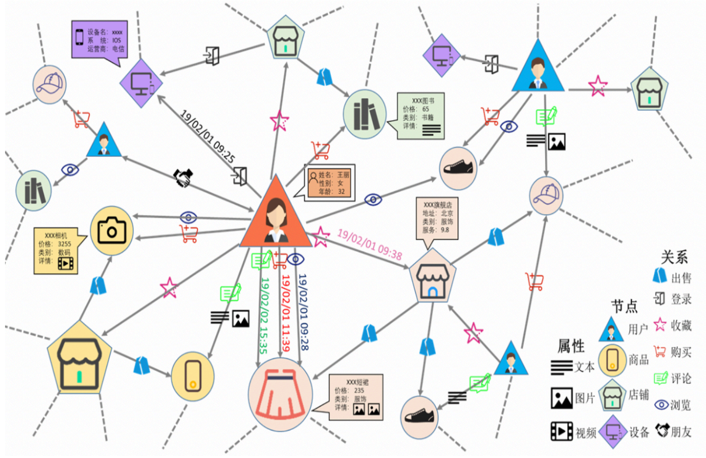



# Heterogeneous Graph

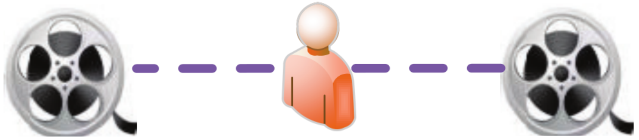
- Multiple types of nodes or links




- Rich semantic information
  - Meta-path: a relation sequence connecting two objects (e.g., Movie-Actor-Movie).

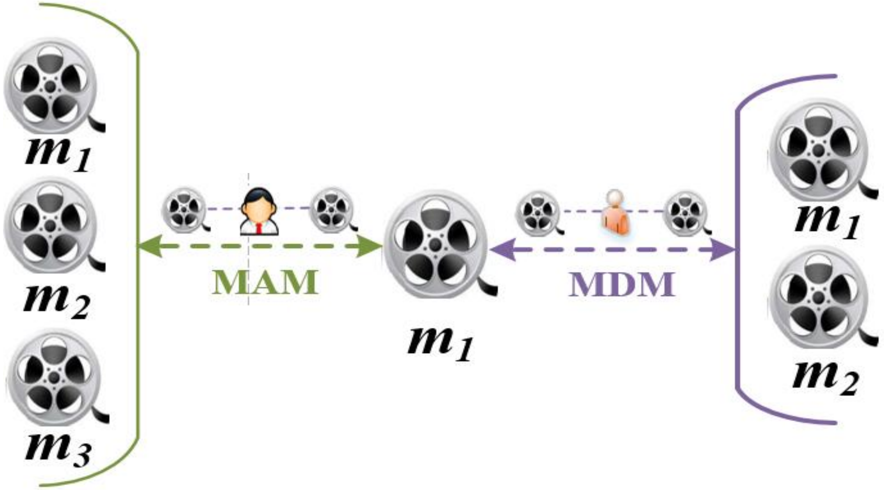


Movie-Director-Moive

Movie-Actor-Moive

Two movies directed by the same director.

Two movies are starred by the same actor.

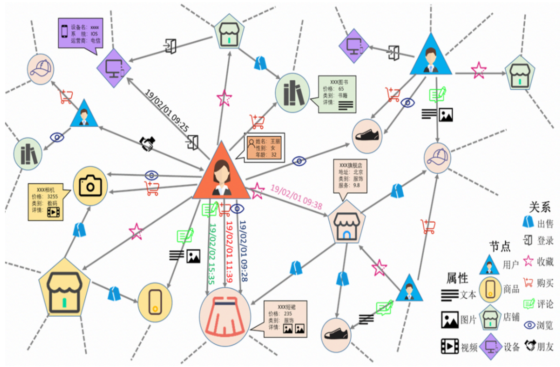


# Homogeneous GNN V.S. Heterogeneous GNN

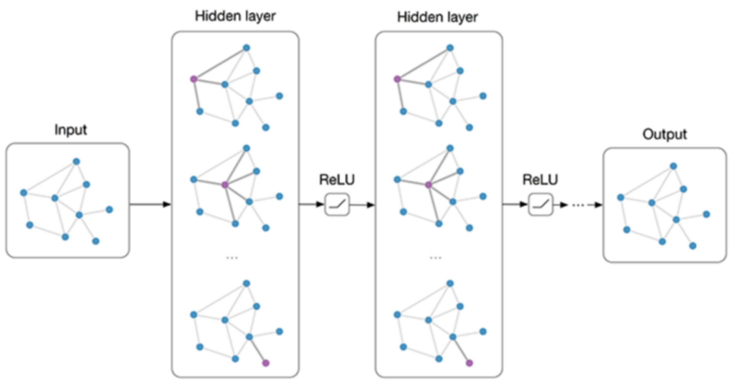
Homogeneous Graph

Complex Interactions
Rich Semantics

Heterogeneous Graph

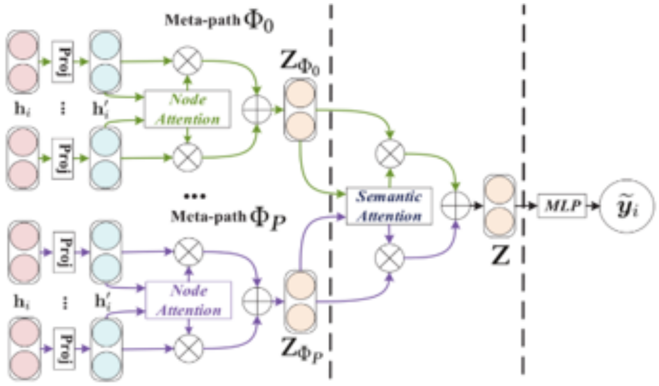


Homogeneous GNN

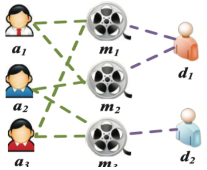



Diverse Inputs
Exquisite Models

Heterogeneous GNN

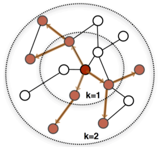


## Development History



- HAN [Wang et al.]
- HeteGNN [Zhang et al.]

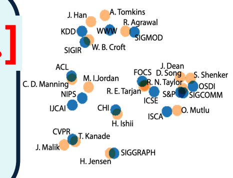
**Heterogeneous GNN**



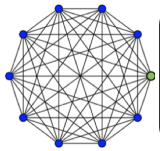
- GCN [Kipf et al.]
- GAT
- Grap

**Homogeneous GNN**

2017~2019



- Metapath2vec [Dong et al.]
- DeepWalk [Bryan et al.]
- SDNE [Wang et al.]

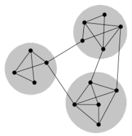


- PathSim [Sun et al.]
- Social Networks [Kwak et al.]

**Heterogeneous Graph Analysis**

- Evolution [Leskovec et al.]
- Social Influence [Tang et al.]

2010~2014

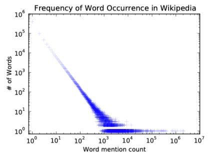


- Influence Max [Domingos et al.]
- Community
 & Newman

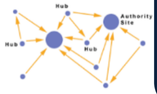
**Homogeneous Graph Analysis**

5~2009

2000~2004




- Small Worlds [Watts et al.]
- Power Law [Faloutsos et al.]



- PageRank [Page et al.]
- HITS [Kleinberg]

1998-1999

1997

# Challenges of Heterogeneous Graph Neural Network

## Challenges of Model Design.

- How to handle the heterogeneity of graph?
  - Diverse types of nodes and edges.
- How to design deeper Heterogeneous GNN?
  - Deeper is better.

## Challenges of Practical Application

- How to design Heterogeneous GNN for practical applications.

CONTENTS

## General framework of Heterogeneous Graph Neural Network

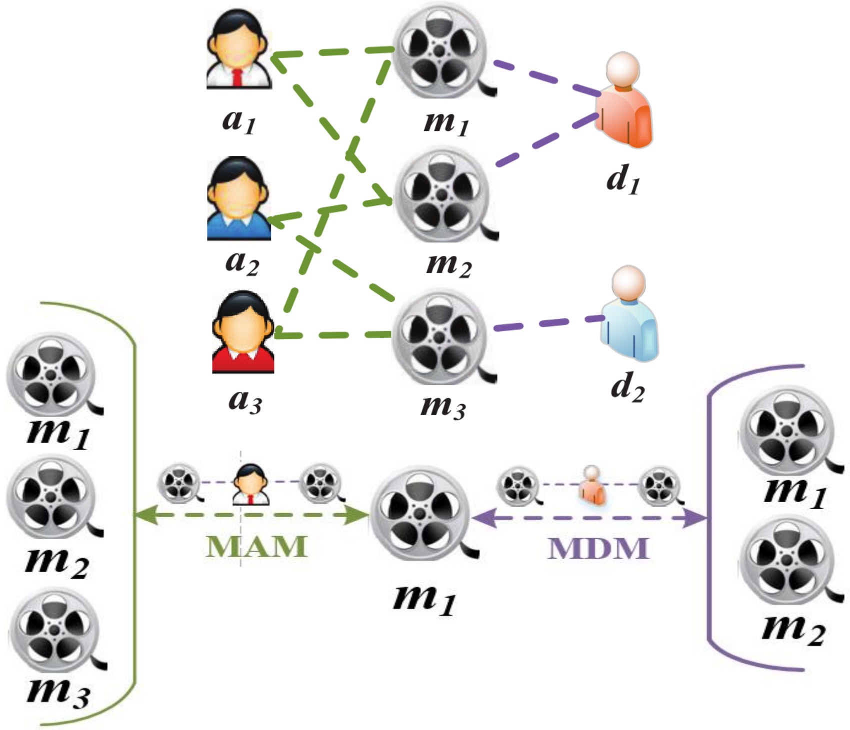
Heterogeneous GNNs usually follow two steps:

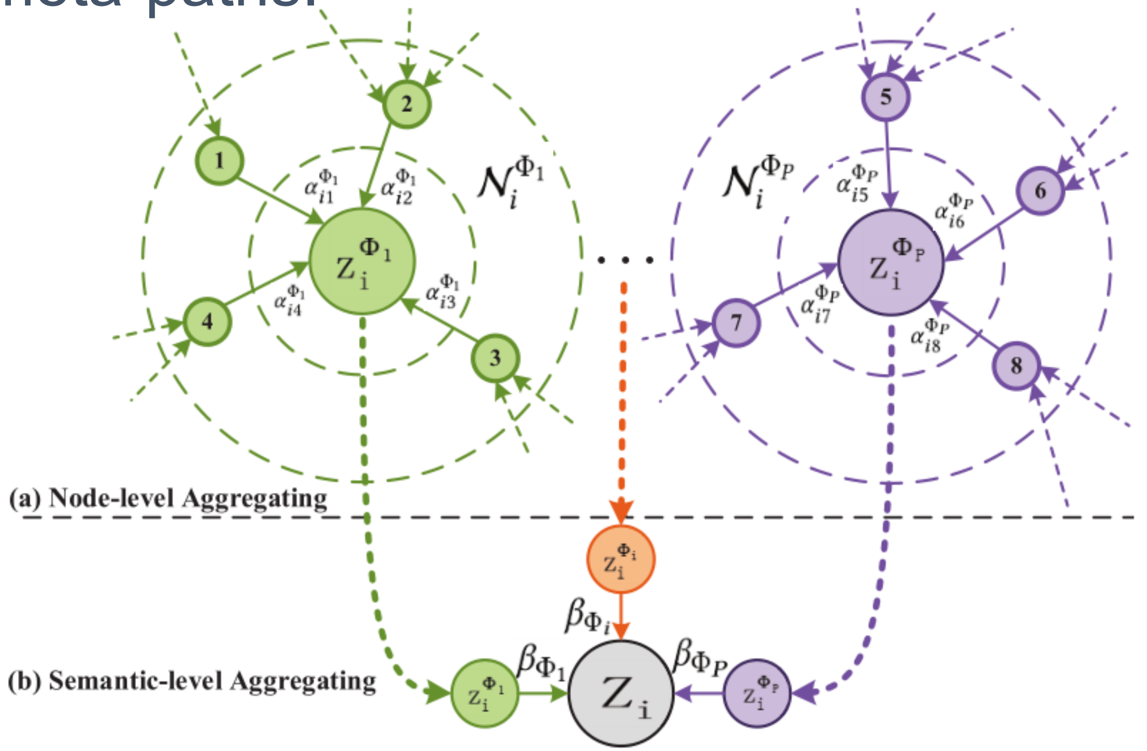
Step1: Node-level Aggregating

- Aggregate neighbors via single meta-path.

Step2: Semantic-level Aggregating

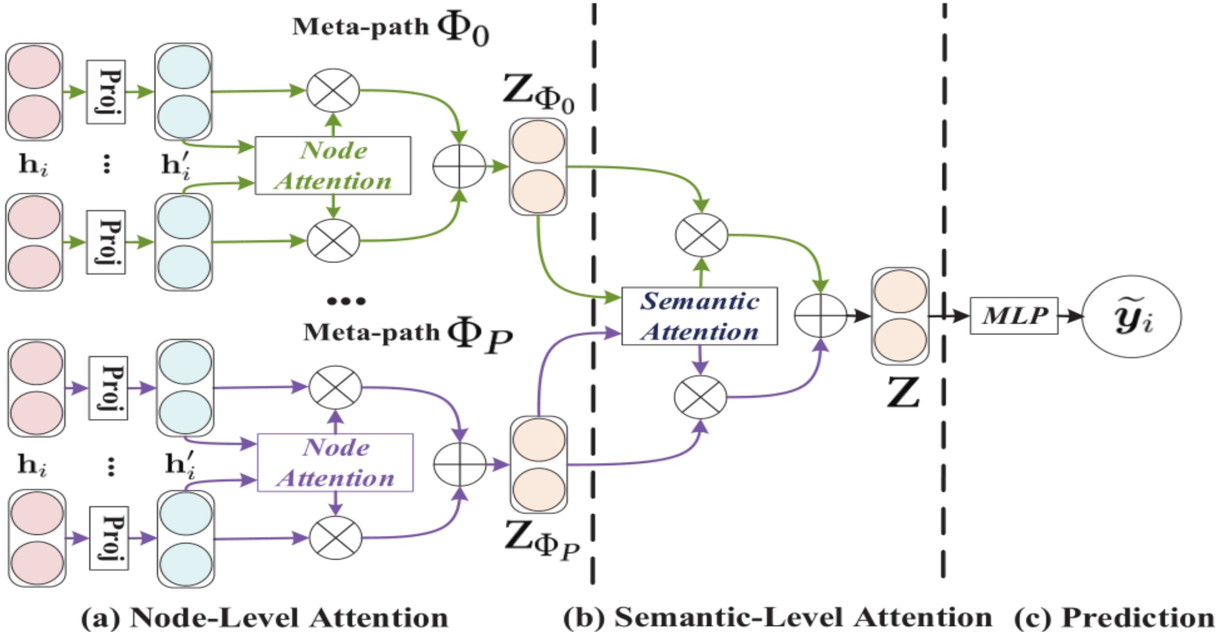
- Aggregate rich semantics via multiple meta-paths.

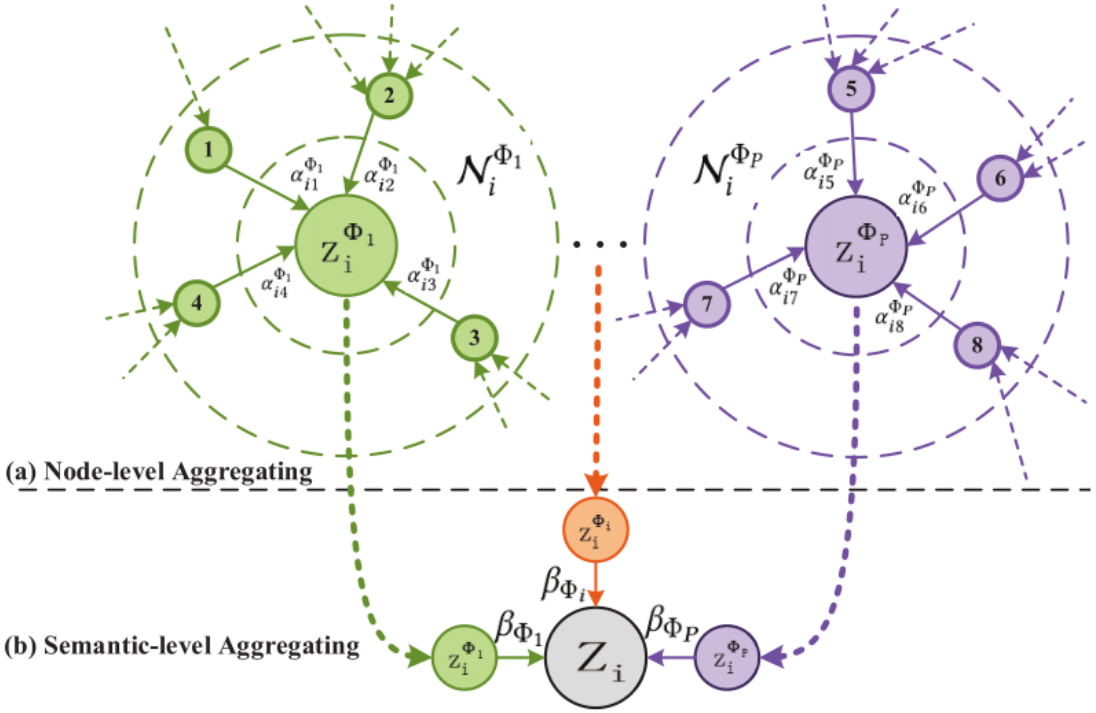


# Models of Heterogeneous GNN

- Heterogeneous Graph Attention Network (HAN)
  - Top-1 Cited Full Paper in WWW2019
- Heterogeneous Graph Propagation Network (HPN)



# Heterogeneous Graph Attention Network

Xiao Wang, Houye Ji
Beijing University of Posts and Telecommunications
Beijing, China
{xiaowang,jhy1993}@bupt.edu.cn

Chuan Shi*, Bai Wang
Beijing University of Posts and Telecommunications
Beijing, China
{shichuan,wangbai}@bupt.edu.cn

Peng Cui, P. Yu
Tsinghua University
Beijing, China
{cuip,psyu}@tsinghua.edu.cn

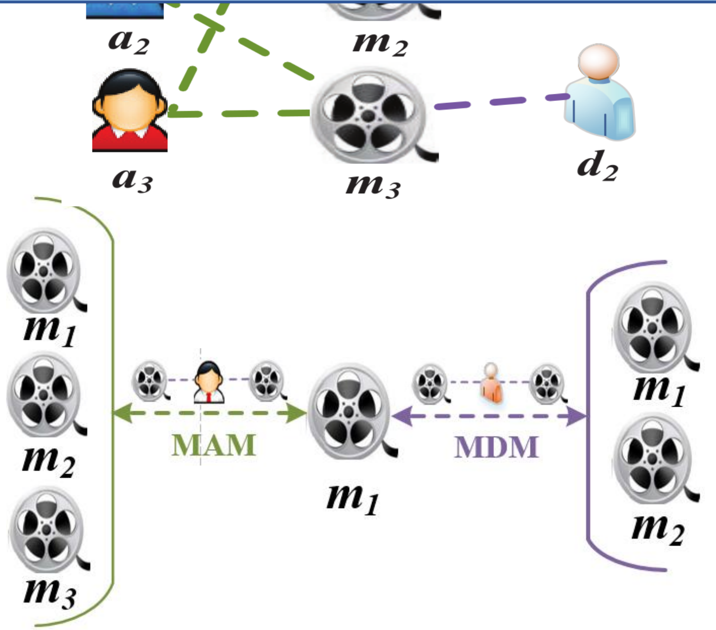
Yanfang Ye
West Virginia University
WV, USA
yanfang.ye@mail.wvu.edu

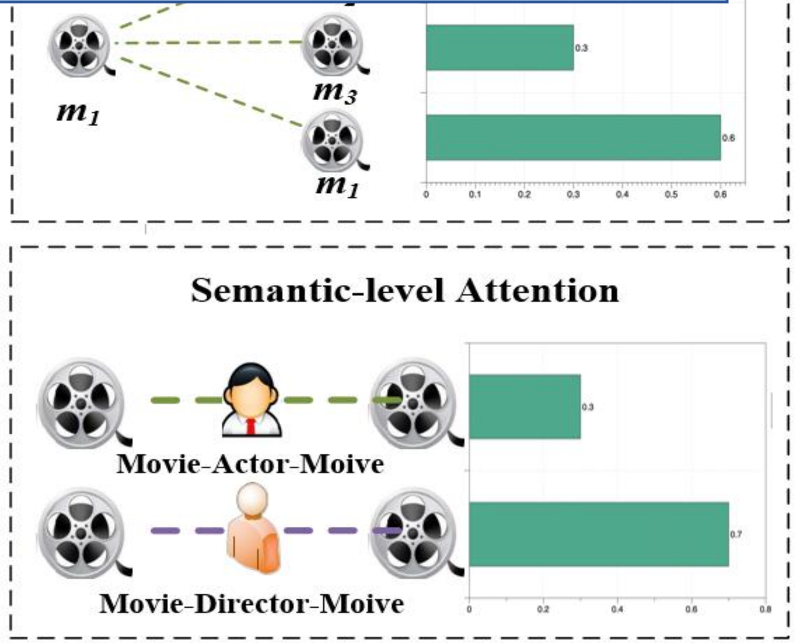
## Challenges

- How to handle the heterogeneity of graph?
- How to discover the differences of meta-path based neighbors?
- How to find some meaningful meta-paths?

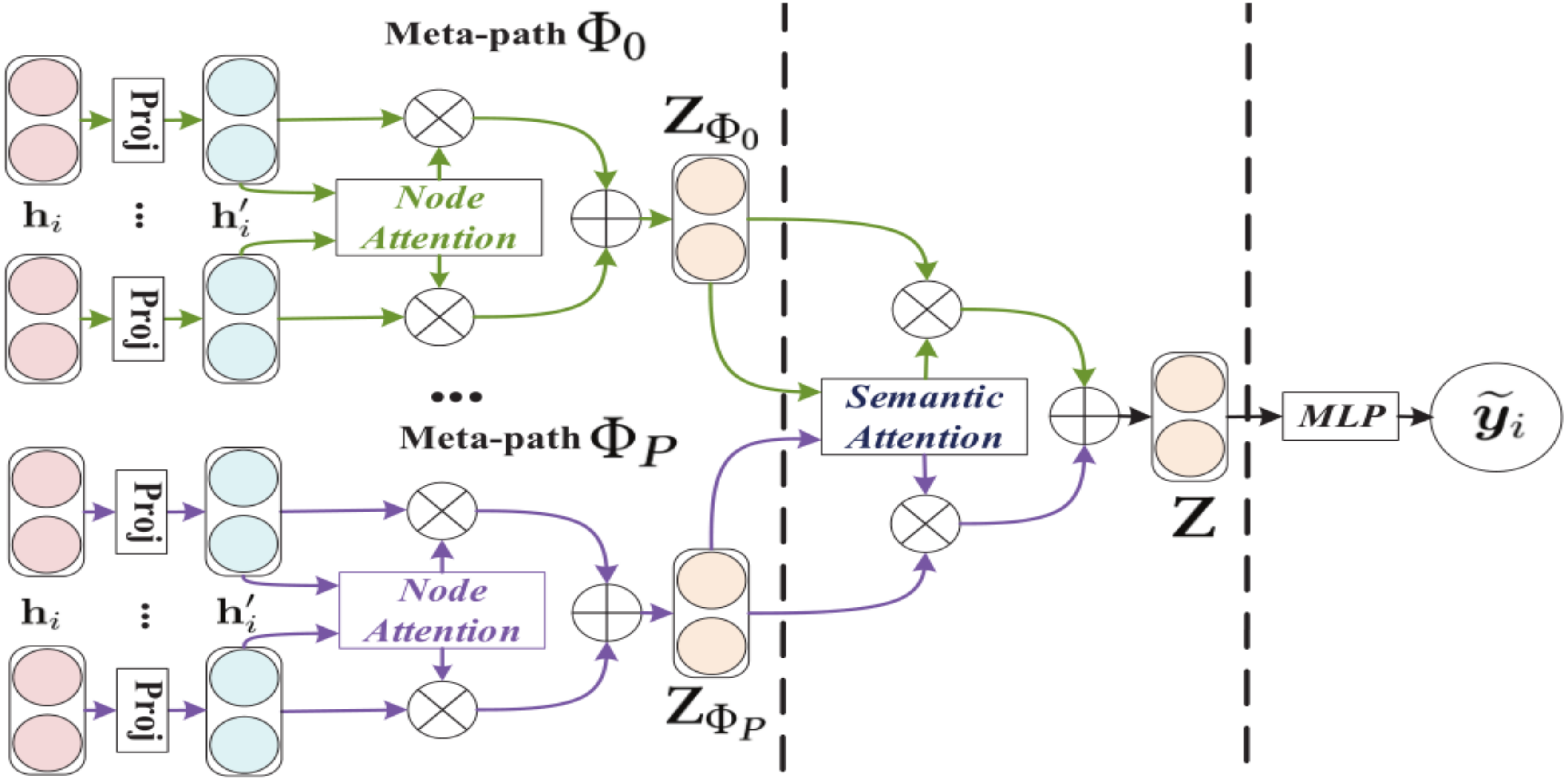
**Our solution:**

**Heterogeneous Graph Attention Network**



# Heterogeneous Graph Attention Network（HAN）



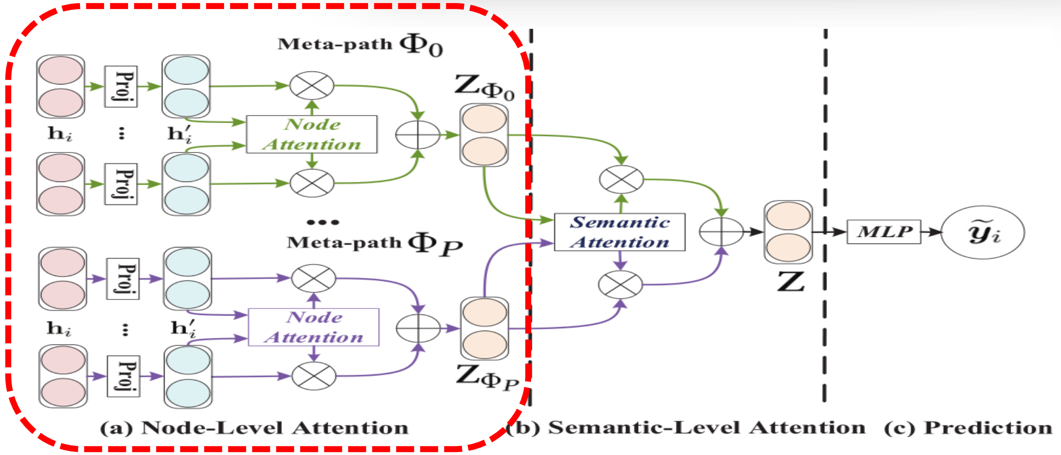

(a) Node-Level Attention (b) Semantic-Level Attention (c) Prediction

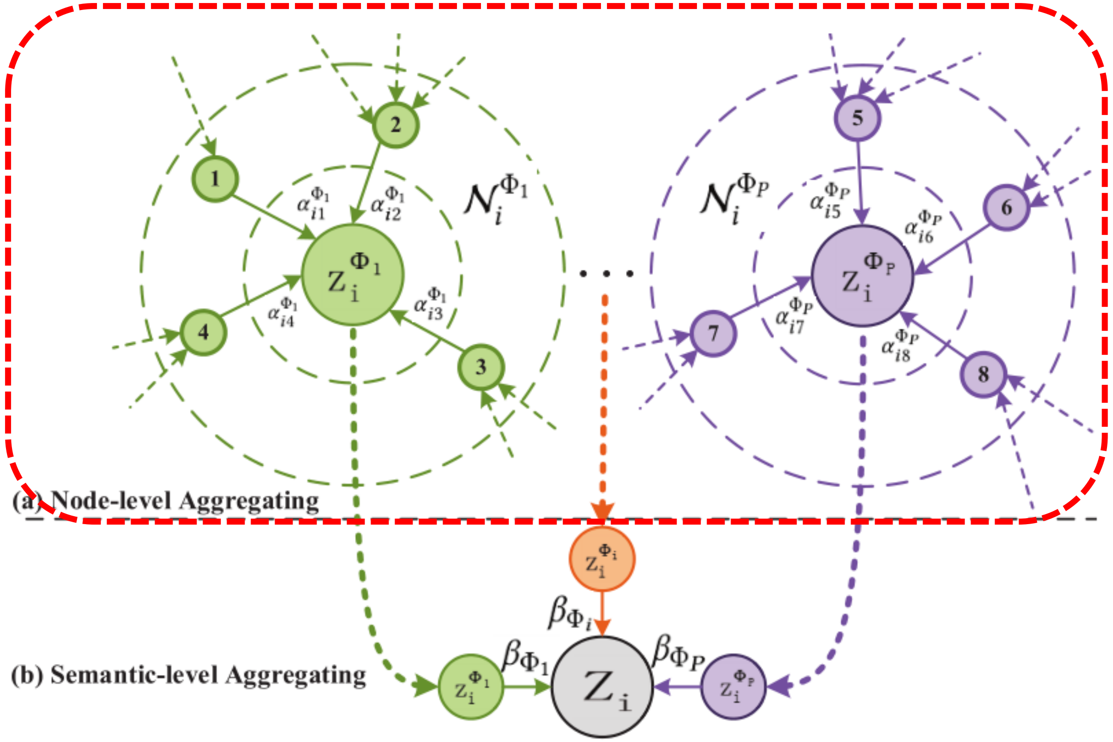
Model heterogeneous structure.

Capture rich semantics.

Task-specific loss.

# Node-Level Attention and Aggregating




- Type-Specific Transformation

$$\mathbf{h}'_i = \mathbf{M}_{\phi_i} \cdot \mathbf{h}_i,$$

Type-specific transformation matrix

- Importance of Neighbors

$$e_{ij}^{\Phi} = att_{node}\left(\mathbf{h}'_i, \mathbf{h}'_j; \Phi\right)$$

$$e_{ij}^{\Phi} = \sigma\left(\mathbf{a}_{\Phi}^{\mathrm{T}} \cdot \left[\mathbf{h}'_i \| \mathbf{h}'_j\right]\right)$$

$$\alpha_{ij}^{\Phi} = softmax_j\left(e_{ij}^{\Phi}\right)$$

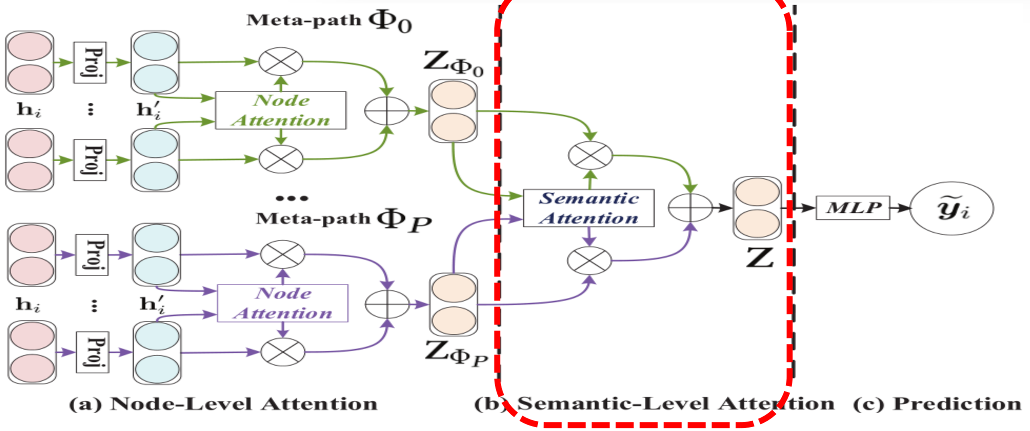
Node-level attention vector

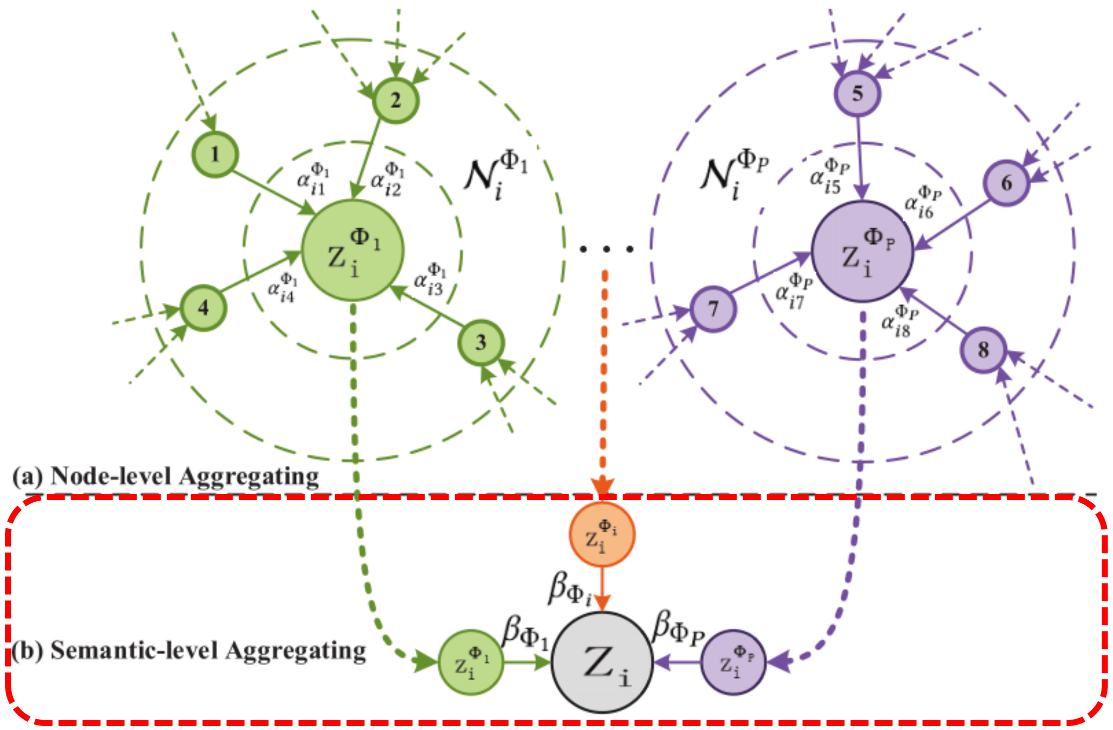
- Node-Level Aggregating

Node weight

$$\mathbf{z}_i^{\Phi} = \sigma\Bigg(\sum_{j \in \mathcal{N}_i^{\Phi}} \alpha_{ij}^{\Phi} \cdot \mathbf{h}'_j\Bigg)$$

# Semantic-Level Attention and Aggregating




- Semantic-Level Attention

$$(\beta_{\Phi_0}, \beta_{\Phi_1}, \ldots, \beta_{\Phi_P}) = att_{sem}(Z_{\Phi_0}, Z_{\Phi_1}, \ldots, Z_{\Phi_P})$$

- Importance of Meta-path

Semantic-level attention vector

$$w_{\Phi_i} = \frac{1}{|\mathcal{V}|} \sum_{i \in \mathcal{V}} \mathbf{q}^{\mathrm{T}} \cdot \tanh(\mathbf{W} \cdot \mathbf{z}_i^{\Phi} + \mathbf{b})$$

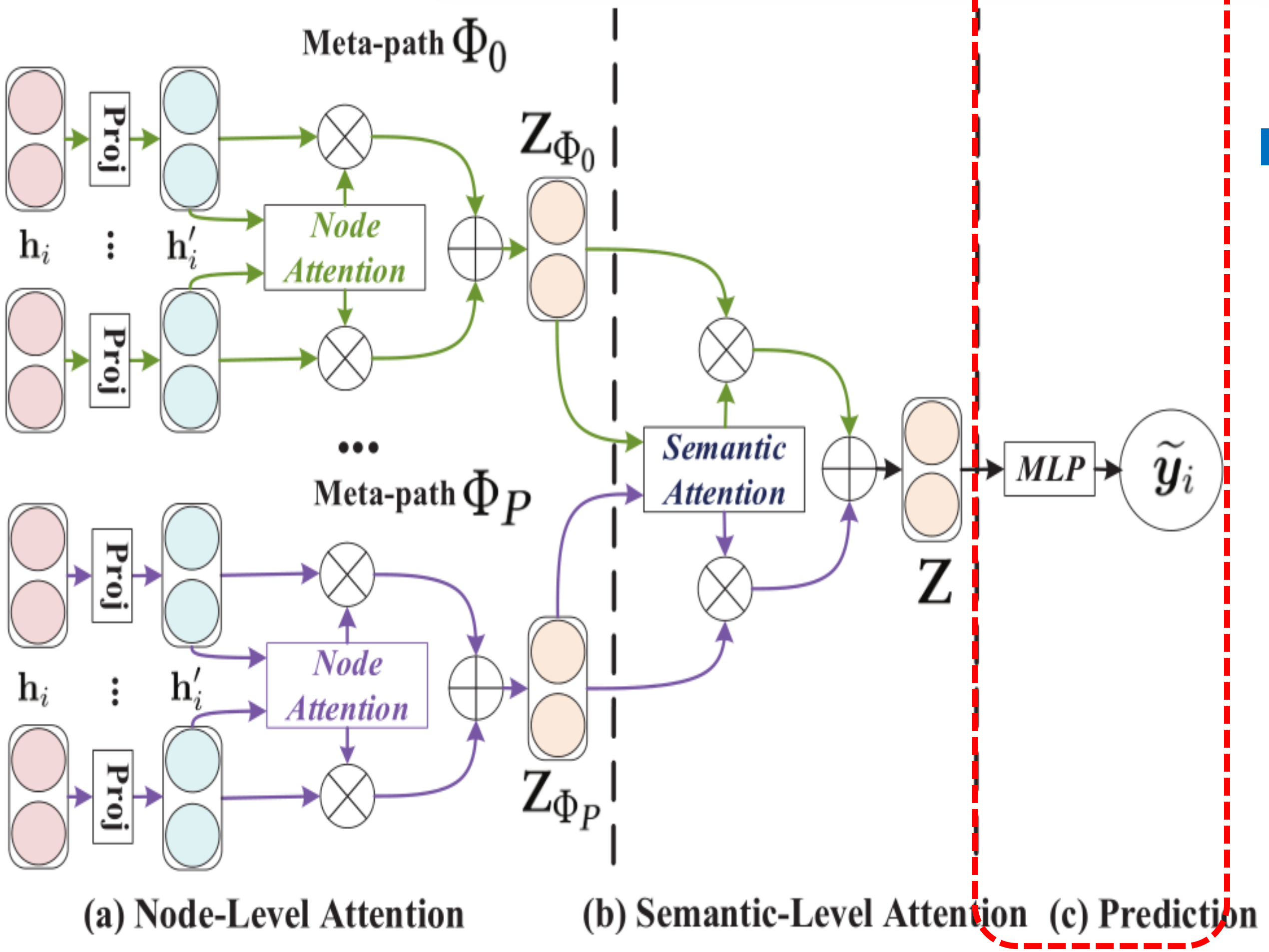
$$\beta_{\Phi_i} = \frac{\exp(w_{\Phi_i})}{\sum_{i=1}^{P} \exp(w_{\Phi_i})}$$

- Semantic-Level Aggregating

$$\mathbf{Z} = \sum_{i=1}^{P} \beta_{\Phi_i} \cdot \mathbf{Z}_{\Phi_i}$$

Semantic weight

## Prediction




Prediction

- Semi-supervised Loss

Parameter of classifier

$$L = -\sum_{l \in \mathcal{Y}_L} \mathrm{Y}^l \ln(\mathrm{C} \cdot \mathrm{Z}^l)$$

Labeled data

**Optimize for the specific task (e.g., node classification).**

# Datasets & Baselines & Tasks

## Baselines

- Deepwalk
- Esim
- Metapath2vec
- HERec
- GCN
- GAT
- $HAN_{nd}$
- $HAN_{sem}$

## Tasks

- Node Classification
- Node Clustering
- Analysis of Attention Mechanism
- Visualization

**Table 2: Statistics of the datasets.**

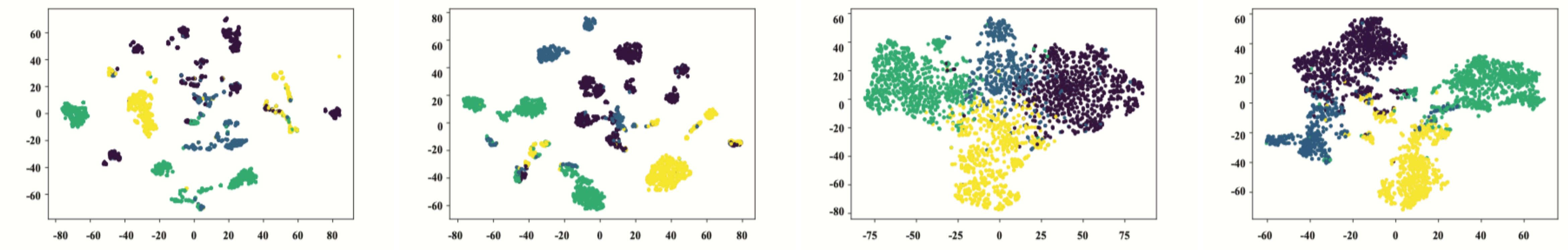
| Dataset | Relations(A-B) | Number of A | Number of B | Number of A-B | Feature | Training | Validation | Test | Meta-paths |
|---|---|---|---|---|---|---|---|---|---|
| DBLP | Paper-Author | 14328 | 4057 | 19645 | 334 | 800 | 400 | 2857 | *APA* |
| | Paper-Conf | 14328 | 20 | 14328 | | | | | *APCPA* |
| | Paper-Term | 14327 | 8789 | 88420 | | | | | *APTPA* |
| IMDB | Movie-Actor | 4780 | 5841 | 14340 | 1232 | 300 | 300 | 2687 | *MAM* |
| | Movie-Director | 4780 | 2269 | 4780 | | | | | *MDM* |
| ACM | Paper-Author | 3025 | 5835 | 9744 | 1830 | 600 | 300 | 2125 | *PAP* |
| | Paper-Subject | 3025 | 56 | 3025 | | | | | *PSP* |

# Node Classification

| Datasets | Metrics | Training | DeepWalk | ESim | metapath2vec | HERec | GCN | GAT | $HAN_{nd}$ | $HAN_{sem}$ | HAN |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ACM | Macro-F1 | 20% | 77.25 | 77.32 | 65.09 | 66.17 | 86.81 | 86.23 | 88.15 | 89.04 | **89.40** |
| | | 40% | 80.47 | 80.12 | 69.93 | 70.89 | 87.68 | 87.04 | 88.41 | 89.41 | **89.79** |
| | | 60% | 82.55 | 82.44 | 71.47 | 72.38 | 88.10 | 87.56 | 87.91 | **90.00** | 89.51 |
| | | 80% | 84.17 | 83.00 | 73.81 | 73.92 | 88.29 | 87.33 | 88.48 | 90.17 | **90.63** |
| | Micro-F1 | 20% | 76.92 | 76.89 | 65.00 | 66.03 | 86.77 | 86.01 | 87.99 | 88.85 | **89.22** |
| | | 40% | 79.99 | 79.70 | 69.75 | 70.73 | 87.64 | 86.79 | 88.31 | 89.27 | **89.64** |
| | | 60% | 82.11 | 82.02 | 71.29 | 72.24 | 88.12 | 87.40 | 87.68 | **89.85** | 89.33 |
| | | 80% | 83.88 | 82.89 | 73.69 | 73.84 | 88.35 | 87.11 | 88.26 | 89.95 | **90.54** |
| DBLP | Macro-F1 | 20% | 77.43 | 91.64 | 90.16 | 91.68 | 90.79 | 90.97 | 91.17 | 92.03 | **92.24** |
| | | 40% | 81.02 | 92.04 | 90.82 | 92.16 | 91.48 | 91.20 | 91.46 | 92.08 | **92.40** |
| | | 60% | 83.67 | 92.44 | 91.32 | 92.80 | 91.89 | 90.80 | 91.78 | 92.38 | **92.80** |
| | | 80% | 84.81 | 92.53 | 91.89 | 92.34 | 92.38 | 91.73 | 91.80 | 92.53 | **93.08** |
| | Micro-F1 | 20% | 79.37 | 92.73 | 91.53 | 92.69 | 91.71 | 91.96 | 92.05 | 92.99 | **93.11** |
| | | 40% | 82.73 | 93.07 | 92.03 | 93.18 | 92.31 | 92.16 | 92.38 | 93.00 | **93.30** |
| | | 60% | 85.27 | 93.39 | 92.48 | 93.70 | 92.62 | 91.84 | 92.69 | 93.31 | **93.70** |
| | | 80% | 86.26 | 93.44 | 92.80 | 93.27 | 93.09 | 92.55 | 92.69 | 93.29 | **93.99** |
| IMDB | Macro-F1 | 20% | 40.72 | 32.10 | 41.16 | 41.65 | 45.73 | 49.44 | 49.78 | **50.87** | 50.00 |
| | | 40% | 45.19 | 31.94 | 44.22 | 43.86 | 48.01 | 50.64 | 52.11 | 50.85 | **52.71** |
| | | 60% | 48.13 | 31.68 | 45.11 | 46.27 | 49.15 | 51.90 | 51.73 | 52.09 | **54.24** |
| | | 80% | 50.35 | 32.06 | 45.15 | 47.64 | 51.81 | 52.99 | 52.66 | 51.60 | **54.38** |
| | Micro-F1 | 20% | 46.38 | 35.28 | 45.65 | 45.81 | 49.78 | 55.28 | 54.17 | 55.01 | **55.73** |
| | | 40% | 49.99 | 35.47 | 48.24 | 47.59 | 51.71 | 55.91 | 56.39 | 55.15 | **57.97** |
| | | 60% | 52.21 | 35.64 | 49.09 | 49.88 | 52.29 | 56.44 | 56.09 | 56.66 | **58.32** |
| | | 80% | 54.33 | 35.59 | 48.81 | 50.99 | 54.61 | 56.97 | 56.38 | 56.49 | **58.51** |

# Node Clustering & Visualization

| Datasets | Metrics | DeepWalk | ESim | metapath2vec | HERec | GCN | GAT | $HAN_{nd}$ | $HAN_{sem}$ | HAN |
|---|---|---|---|---|---|---|---|---|---|---|
| ACM | NMI | 41.61 | 39.14 | 21.22 | 40.70 | 51.40 | 57.29 | 60.99 | 61.05 | **61.56** |
| | ARI | 35.10 | 34.32 | 21.00 | 37.13 | 53.01 | 60.43 | 61.48 | 59.45 | **64.39** |
| DBLP | NMI | 76.53 | 66.32 | 74.30 | 76.73 | 75.01 | 71.50 | 75.30 | 77.31 | **79.12** |
| | ARI | 81.35 | 68.31 | 78.50 | 80.98 | 80.49 | 77.26 | 81.46 | 83.46 | **84.76** |
| IMDB | NMI | 1.45 | 0.55 | 1.20 | 1.20 | 5.45 | 8.45 | 9.16 | 10.31 | **10.87** |
| | ARI | 2.15 | 0.10 | 1.70 | 1.65 | 4.40 | 7.46 | 7.98 | 9.51 | **10.01** |



(a) GCN (b) GAT (c) metapath2vec (d) HAN

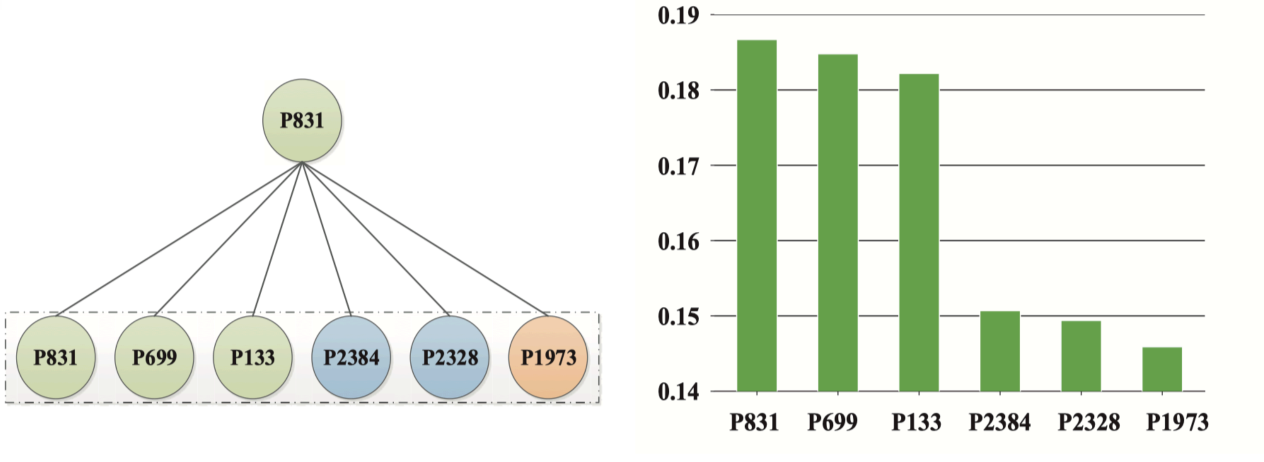
**Figure 6: Visualization embedding on DBLP. Each point indicates one author and its color indicates the research area.**

# Attention Analysis

- Node-Level Attention (e.g., P831[1])

**P831 > P699 > … > P2328 > P1973**

Important neighbors have larger attention values.



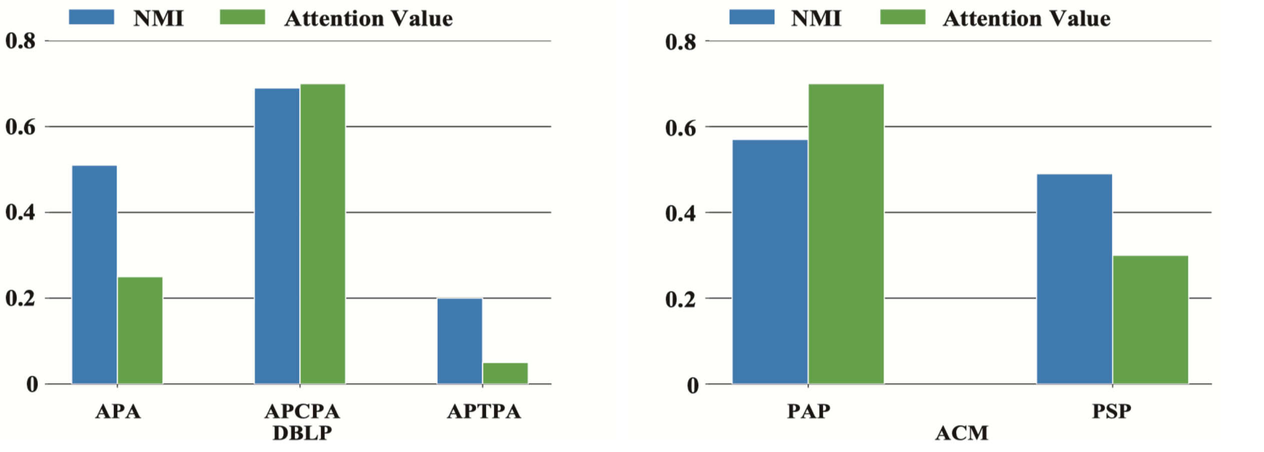

(a) Meta-path based neighbors of P831 (b) Attention values of P831’s neighbors

- Semantic-Level Attention

**APCPA > APA > APTPA**

**PAP > PSP**

Important meta-paths have larger attention values.




(a) NMI values on DBLP (b) NMI values on ACM

Xintao Wu, et al. Screening and Interpreting Multi-item Associations Based on Log-linear Modeling. KDD 2003

## Summary

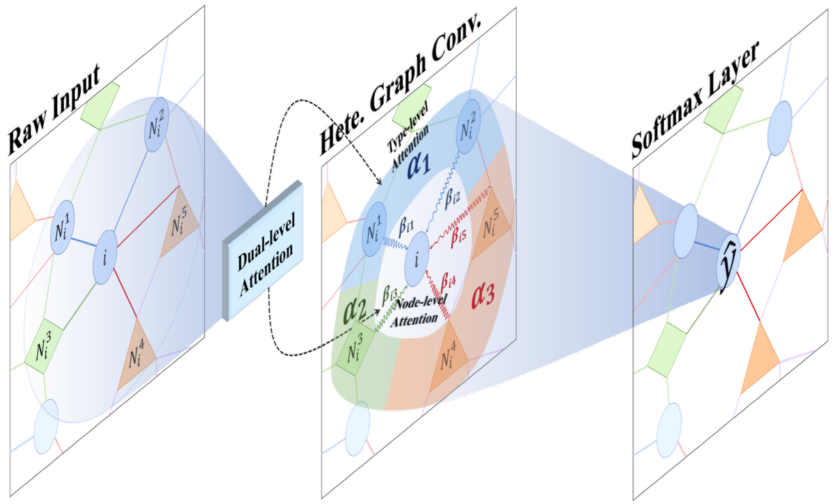
- The first attempt to study the heterogeneous graph neural network based on attention mechanism.
- A novel heterogeneous graph attention network (HAN) which includes both of the node-level and semantic-level attentions.
- The state-of-the-art performance and good interpretability.

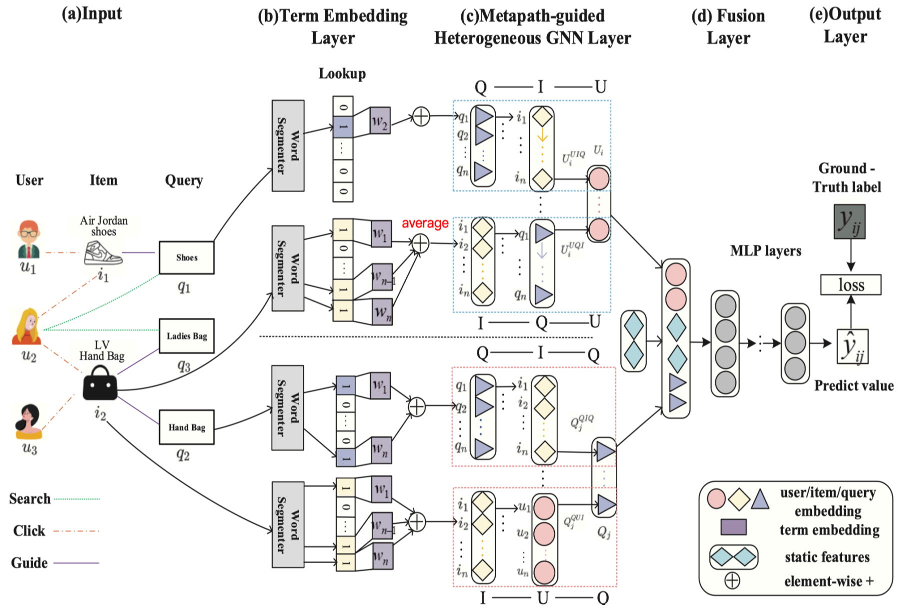
| Method | Node-level Aggregating | Semantic-level Aggregating | Task |
|---|---|---|---|
| HAN | Attention | Attention | Node classification/Clustering |

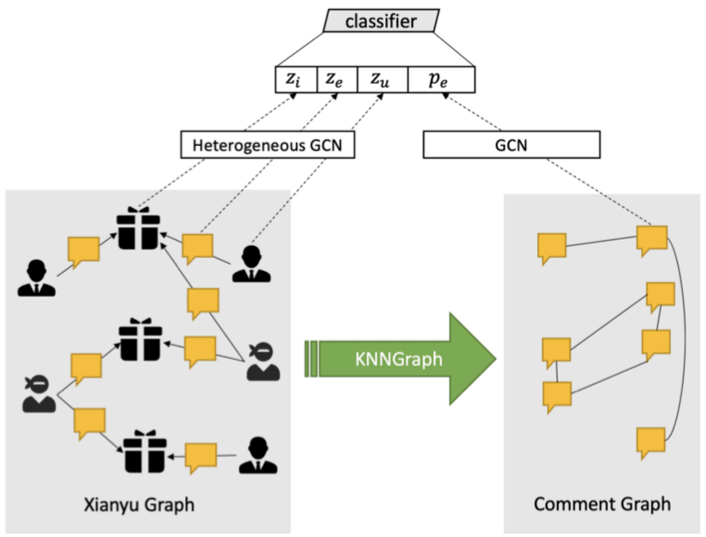
CONTENTS

# Applications of Heterogeneous GNN

- Heterogeneous Graph Attention Networks for Semi-supervised Short Text Classification
- Metapath-guided Heterogeneous Graph Neural Network for Intent Recommendation
- Spam Review Detection with Graph Convolutional Networks



# Heterogeneous Graph Attention Networks for Semi-supervised Short Text Classification

**Linmei Hu[1], Tianchi Yang[1], Chuan Shi[*1], Houye Ji[1], Xiaoli Li[2]**
[1]Beijing University of Posts and Telecommunications, China
[2]Nanyang Technological University, Singapore
{hulinmei,yangtianchi,shichuan,jhy1993}@bupt.edu.cn
xlli@i2r.a-star.edu.sg

## Short texts

**Tweets**

The hard truth about the United States is that the money other countries spend on health and infrastructure, we spend on war.

**Reviews**

★★★★★ Home secured.
By M. Wolff on January 25, 2014
I purchased this along with 4 turtles and a rat.

18 years worth of karate lessons later, I finally feel safe to leave my house at night.

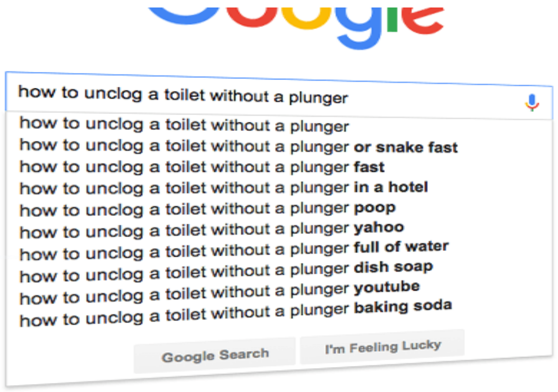
**News**

Trump claims he's suing 'various people' for violating confidentiality agreements
By Veronica Stracqualursi and Pamela Brown, CNN
Updated 4:37 PM ET, Sat August 31, 2019

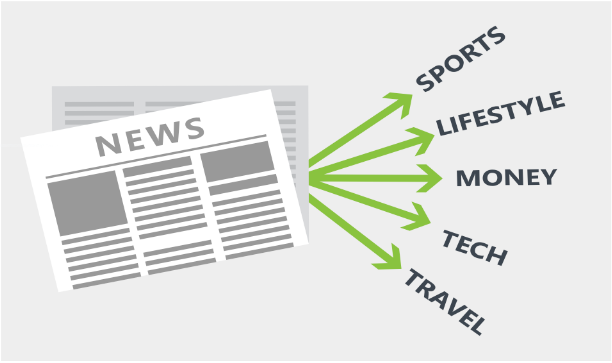
**Queries**

how to use google effectively?

## Short texts classification




Query Intent Classification



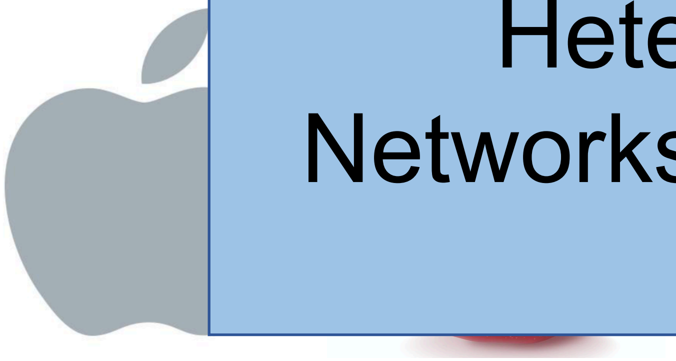

New Categorization

## Challenges of short texts classification

- How to deal with semantically sparse and ambiguous.
- How to utilize limited labeled training data.
- How to discover different importance of different information?

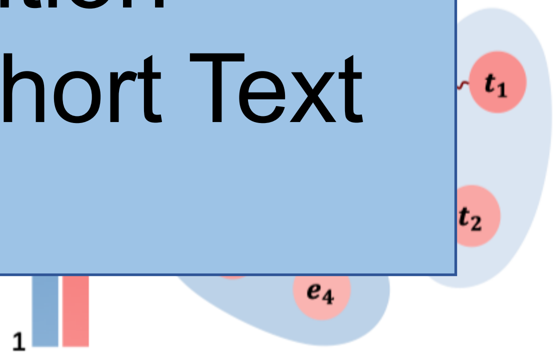
**Our solution:**

**Heterogeneous Graph Attention Networks for Semi-supervised Short Text Classification**



Ambiguous

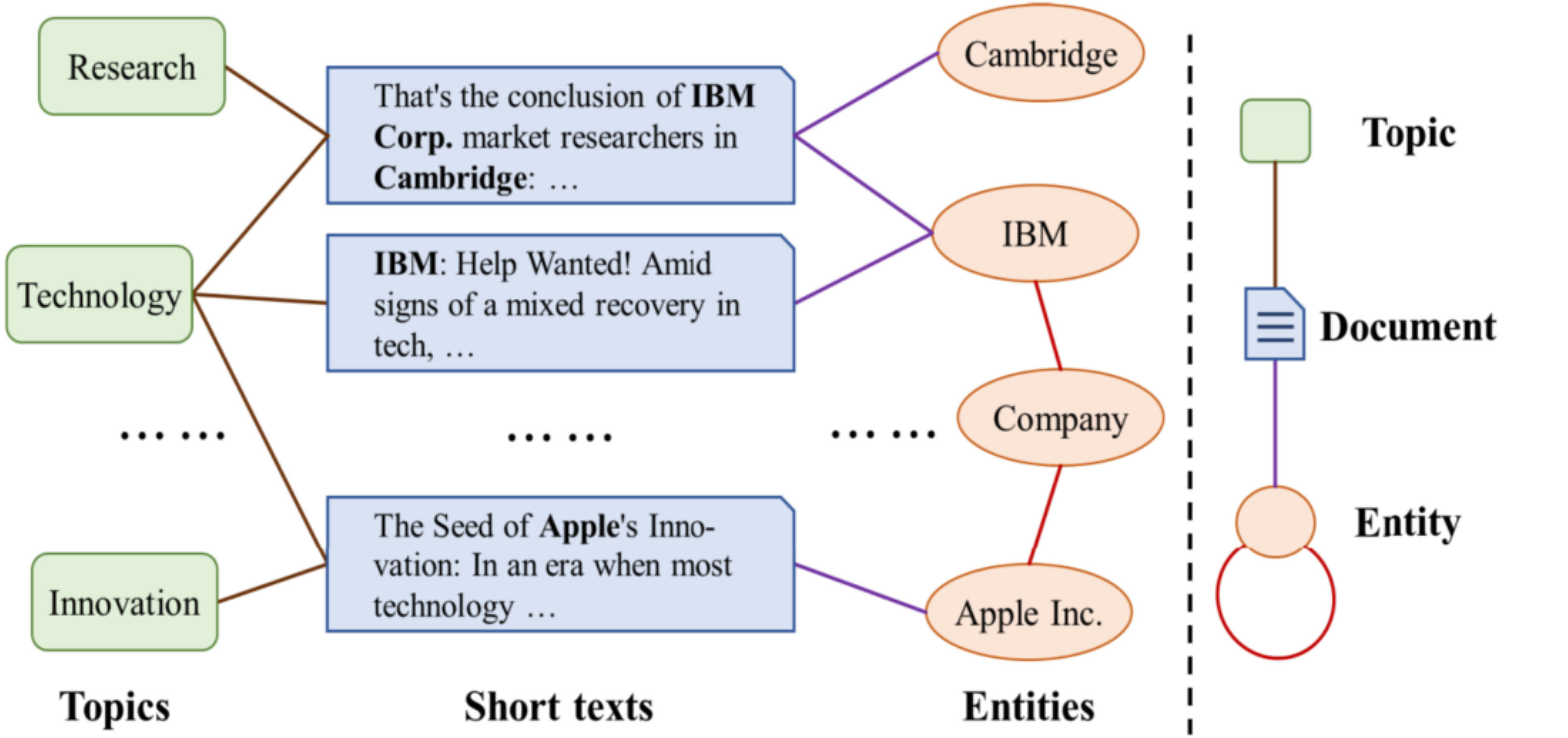
Limited Label



Different Importance

## HGAT for Semi-supervised Short Text Classification

### Heterogeneous Graph for Short Texts

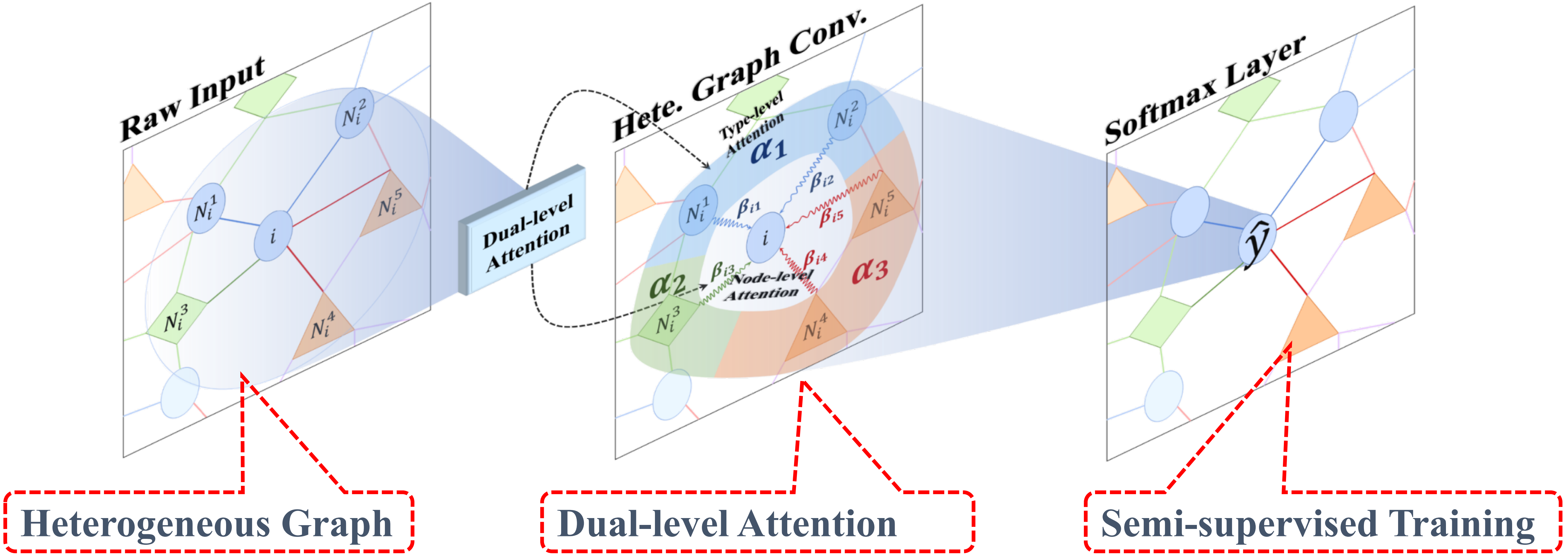



(a) HIN for short texts (b) HIN schema

- **Nodes**
  - **Text**: Document in the corpus.
  - **Topic**: Mined by LDA.
  - **Entity**: Recognized by an entity linking tool, named TagMe.
- **Edges**
  - **Text-Topic**: The text is assigned to the Top *P* topics.
  - **Text-Entity**: The text contains the entity.
  - **Entity-Entity**: The similarity score is above threshold.

## HGAT for Semi-supervised Short Text Classification

# Heterogeneous Graph ATtention Network (HGAT)



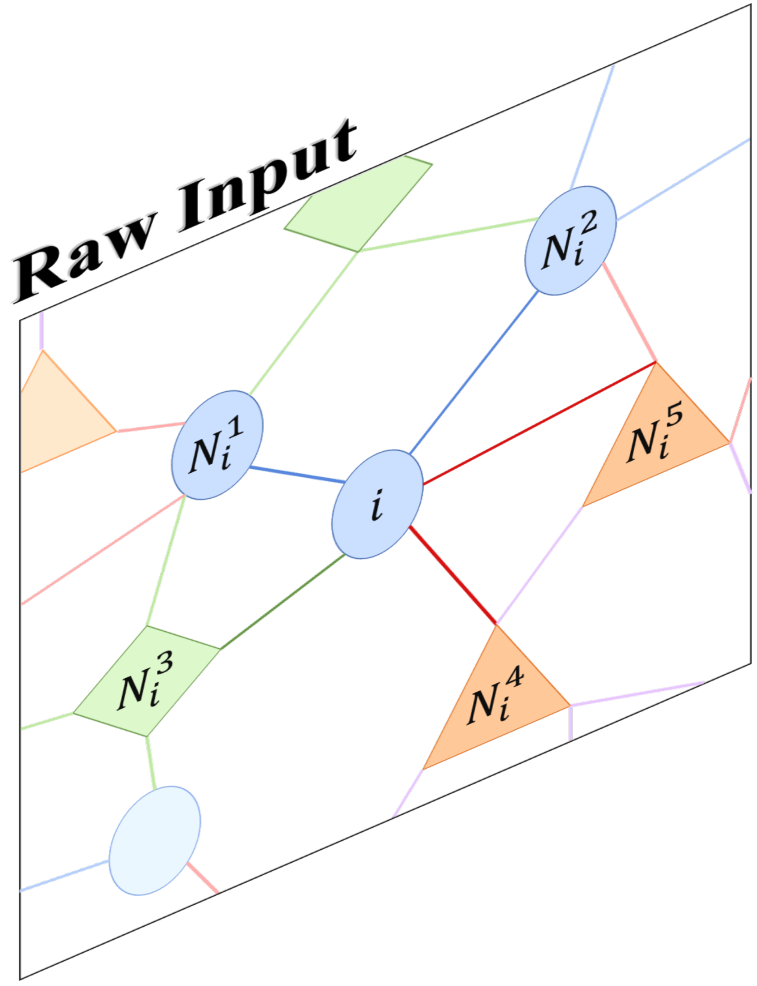
## HGAT for Semi-supervised Short Text Classification

### Raw Input

The **HIN** $\mathcal{G} = (\mathcal{V}, \mathcal{E})$;

The node features for

- **Texts**:
  TF-IDF vectors of the contents.
- **Topics**:
  Word distributions on vocabulary.
- **Entities**:
  TF-IDF vectors of entity descriptions.



## HGAT for Semi-supervised Short Text Classification

# Heterogeneous Graph Convolution

**Heterogeneous Graph**

$$H^{(l+1)} = \sigma\left(\sum_{\tau\in\mathcal{T}} \tilde{A}_\tau \cdot H_\tau^{(l)} \cdot W_\tau\right)$$

**Type-level Attention**

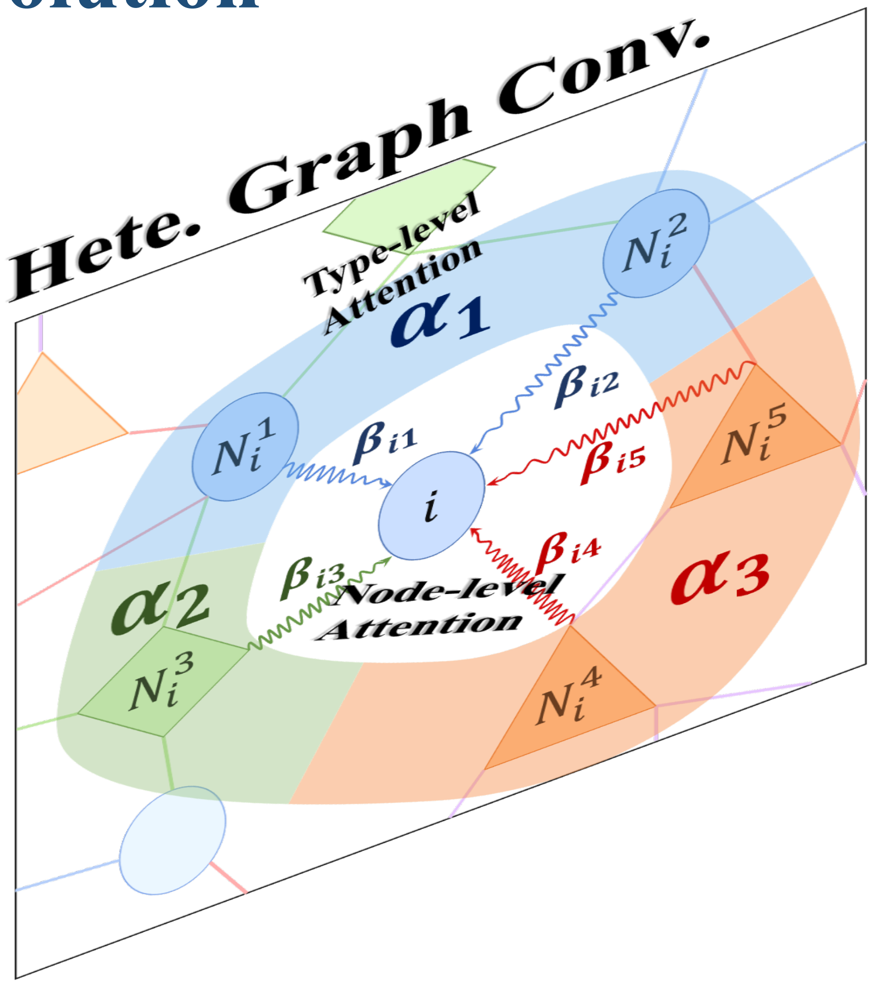
$$a_\tau = \sigma(\mu_\tau^T \cdot [h_v || h_\tau]), \quad \alpha_\tau = \frac{\exp(a_\tau)}{\sum_{\tau'\in\mathcal{T}} \exp(a_{\tau'})}.$$

**Node-level Attention**

$$b_{vv'} = \sigma(\nu^T \cdot \alpha_{\tau'}[h_v || h_{v'}]) \; \beta_{vv'} = \frac{\exp(b_{vv'})}{\sum_{i\in\mathcal{N}_v} \exp(b_{vi})}.$$

**Dual-level Attention Based Hete. Graph Conv.**

$$H^{(l+1)} = \sigma\left(\sum_{\tau\in\mathcal{T}} \mathcal{B}_\tau \cdot H_\tau^{(l)} \cdot W_\tau\right)$$



## HGAT for Semi-supervised Short Text Classification

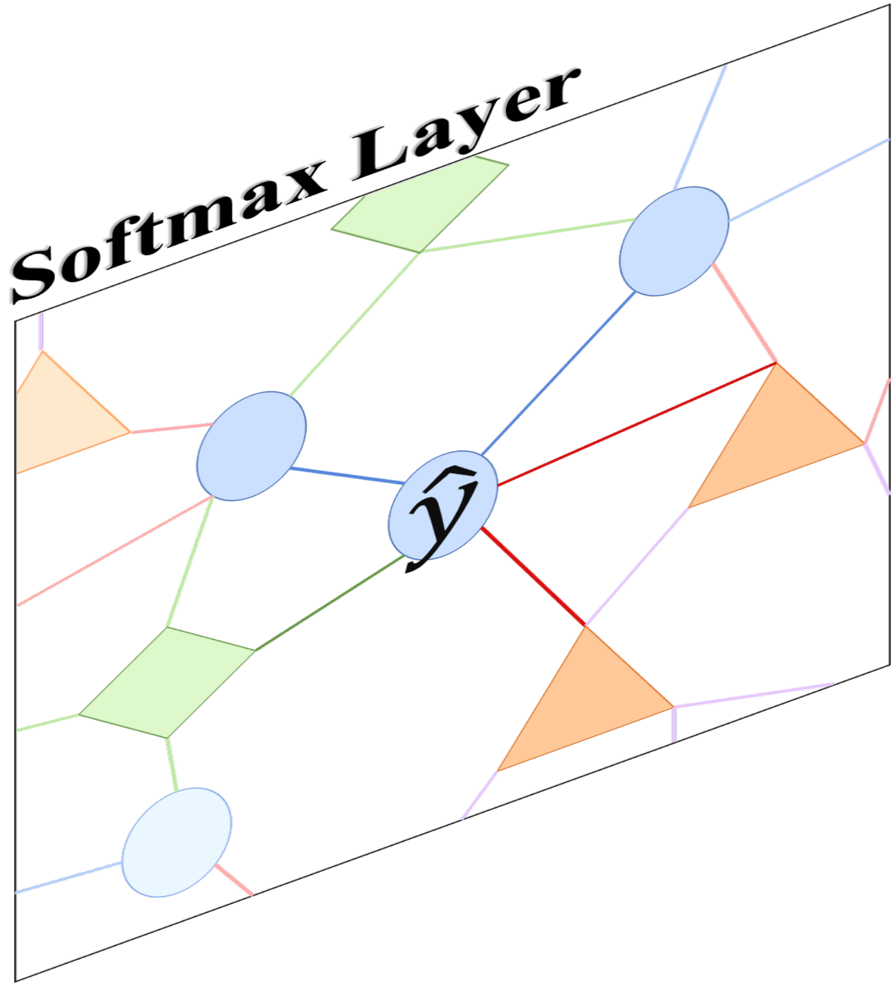
### SoftMax Layer

**SoftMax Layer**

$$Z = \mathrm{softmax}\left(H^{(L)}\right)$$

**Loss Function**

$$\mathcal{L}_{\text{loss}} = \sum_{i \in D_{\text{train}}} \sum_{j=1}^{C} Y_{ij} \log Z_{ij} + \eta \|\Theta\|_2$$



## HGAT for Semi-supervised Short Text Classification

| | #docs | #tokens | #entities | #classes |
|---|---|---|---|---|
| AGNews | 6,000 | 18.4 | 0.9 (72%) | 4 |
| Snippets | 12,340 | 14.5 | 4.4 (94%) | 8 |
| Ohsumed | 7,400 | 6.8 | 3.1 (96%) | 23 |
| TagMyNews | 32,549 | 5.1 | 1.9 (86%) | 7 |
| MR | 10,662 | 7.6 | 1.8 (76%) | 2 |
| Twitter | 10,000 | 3.5 | 1.1 (63%) | 2 |

Table 1: Statistics of the datasets.

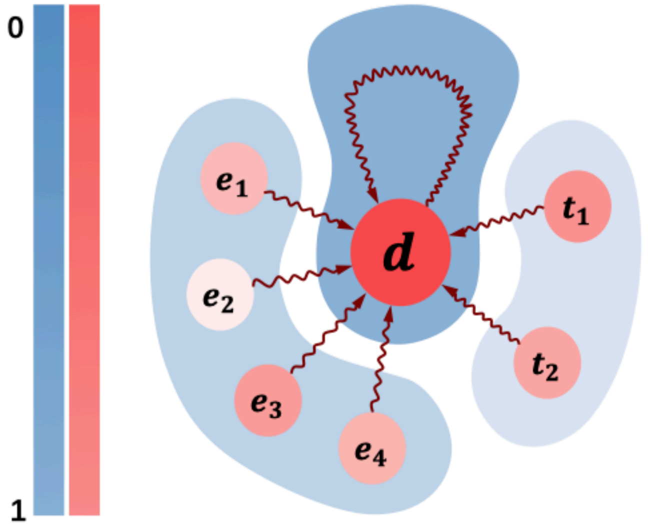
## HGAT for Semi-supervised Short Text Classification

| Dataset | GCN -HIN | HGAT w/o ATT | HGAT -Type | HGAT -Node | HGAT |
|---|---|---|---|---|---|
| AGNews | 70.87 | 70.97 | 71.54 | 71.76 | **72.10*** |
| Snippets | 76.69 | 80.42 | 81.68 | 81.93 | **82.36*** |
| Ohsumed | 40.25 | 41.31 | 41.95 | 42.17 | **42.68*** |
| TagMyNews | 56.33 | 59.41 | 60.78 | 61.29 | **61.72*** |
| MR | 60.81 | 62.13 | 62.27 | 62.31 | **62.75*** |
| Twitter | 61.59 | 62.35 | 62.95 | 62.45 | **63.21*** |

Table 3: Test accuracy (%) of our variants.

**All modules are efficient.**

## HGAT for Semi-supervised Short Text Classification




**Short Text $d$**

**Shawn Green (Entity $e_1$)** hit two home runs, as **Los Angeles (Entity $e_2$)** defeated the **Atlanta Braves (Entity $e_3$)** 7-4 in a battle of National League division leaders at **Dodger Stadium (Entity $e_4$)**.

**Topic $t_1$:**

| | | | | |
|---|---|---|---|---|
| game | sox | red | beat | team |
| clubs | season | win | astros | run |

**Topic $t_2$:**

| | | | | |
|---|---|---|---|---|
| wins | awards | prix | star | prize |
| greek | china | grand | british | olympics |

Figure 5: Visualization of the dual-level attention including node-level attention (shown in red) and type-level attention (shown in blue). Each topic $t$ is represented by top 10 words with highest probabilities.

## Summary

- The first present a flexible HIN framework for modeling the short texts.
- A novel model HGAT with a dual-level attention mechanism including node-level and type-level attention.
- The state-of-the-art performance and good interpretability.

| Method | Node-level Aggregating | Semantic-level Aggregating | Task |
|---|---|---|---|
| HGAT | Attention | Attention | Short texts classification |

# Metapath-guided Heterogeneous Graph Neural Network for Intent Recommendation

Shaohua Fan
Beijing University of Posts and Telecommunications
Beijing, China
fanshaohua92@163.com

Junxiong Zhu
Alibaba Group
Hangzhou, China
xike.zjx@taobao.com

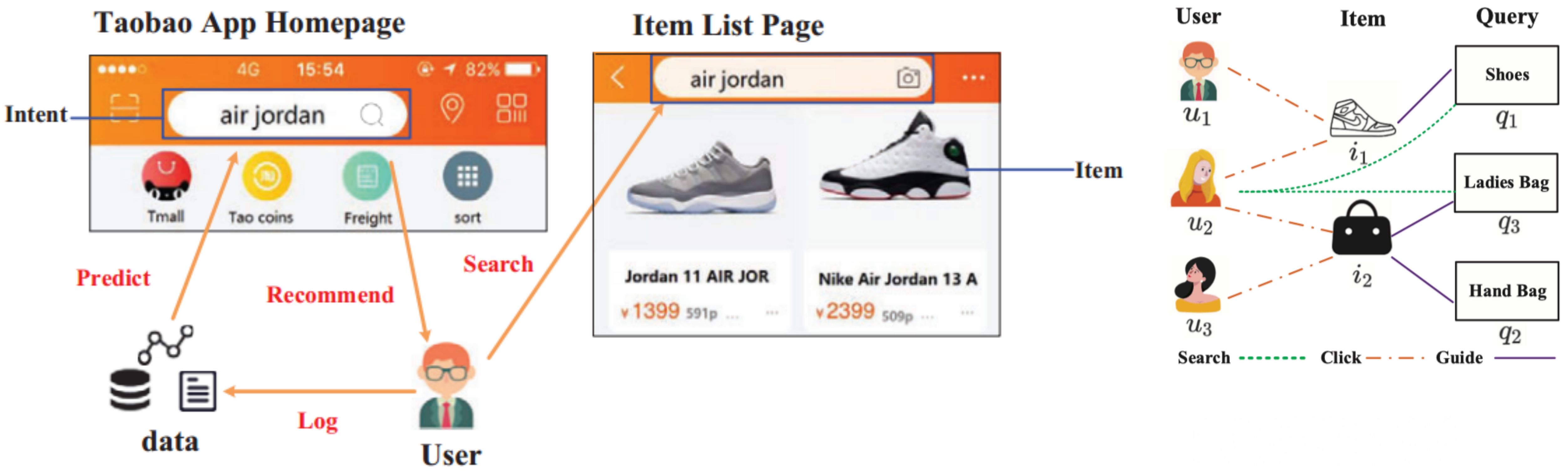
Xiaotian Han
Beijing University of Posts and Telecommunications
Beijing, China
hanxiaotian.h@gmail.com

Chuan Shi*, Linmei Hu
Beijing University of Posts and Telecommunications
Beijing, China
shichuan@bupt.edu.cn,
hulinmei1991@gmail.com

Biyu Ma
Alibaba Group
Hangzhou, China
biyu.mby@alibaba-inc.com

Yongliang Li
Alibaba Group
Hangzhou, China
anthonylee.liyl@tmall.com

## Intent Recommendation at Taobao



## Challenges of Intent Recommendation at Taobao

- **Heterogeneity**

  Comprehensively and flexibly utilize heterogeneous information

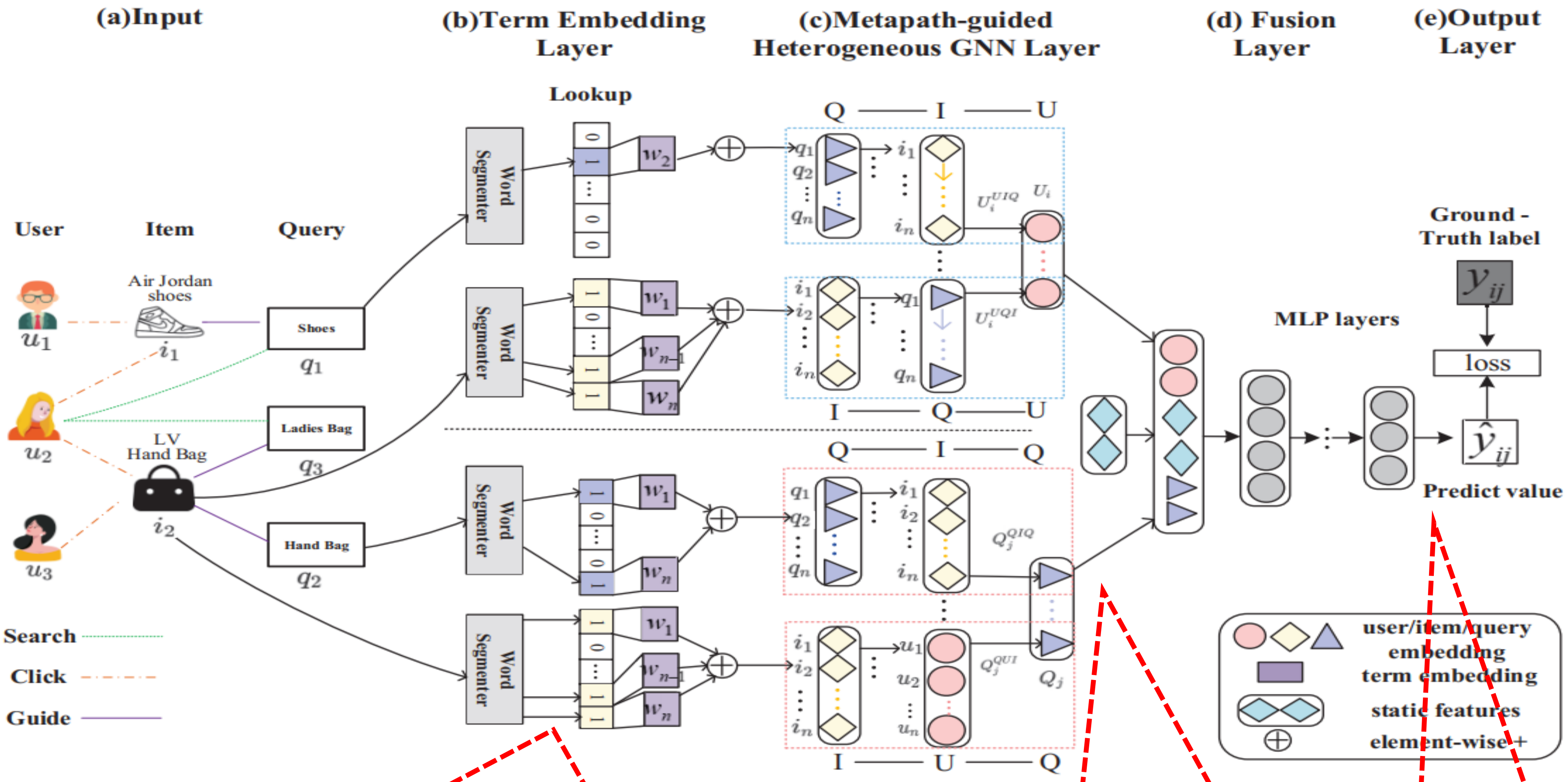
Our solution:

Metapath-guided Heterogeneous Graph Neural Network for Intent Recommendation

- **Task-guided**

  A more useful classification model for intent recommendation

# Metapath-guided Heterogeneous Graph Neural Network for Intent Recommendation

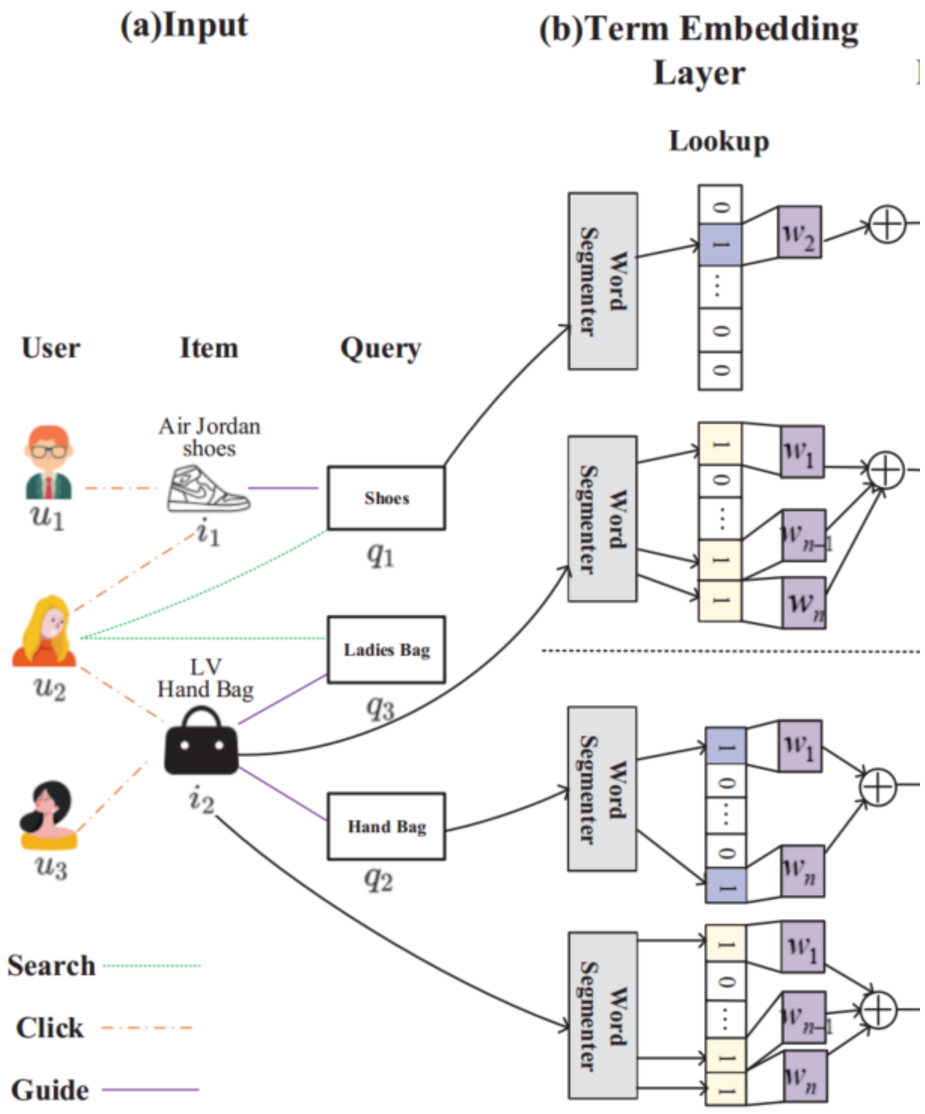



Handle large and Dynamic data

Capture rich semantics.

Task-specific loss.

# Uniform term embedding




## Parameter share

- Queries and items are constituted by the same term embedding

$$
\begin{array}{lccccc}
 & \{w_1, & w_2, & \cdots, & w_{n-1}, & w_n\} \\
q_2 = & (1, & 0, & \cdots, & 0, & 1) \\
i_2 = & (1, & 0, & \cdots, & 1, & 1).
\end{array}
$$

- Reduce parameter space complexity

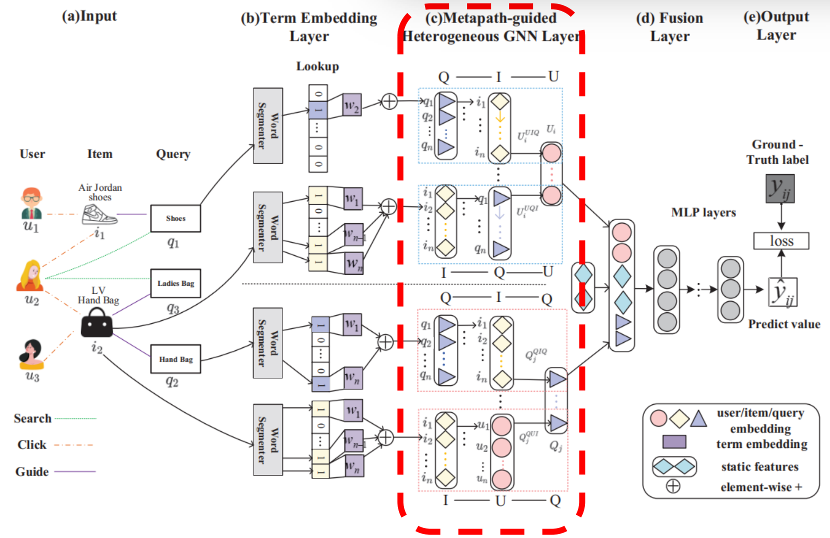
Traditional latent factor model: $O(d*M+n*h) \approx O(d*M)$.

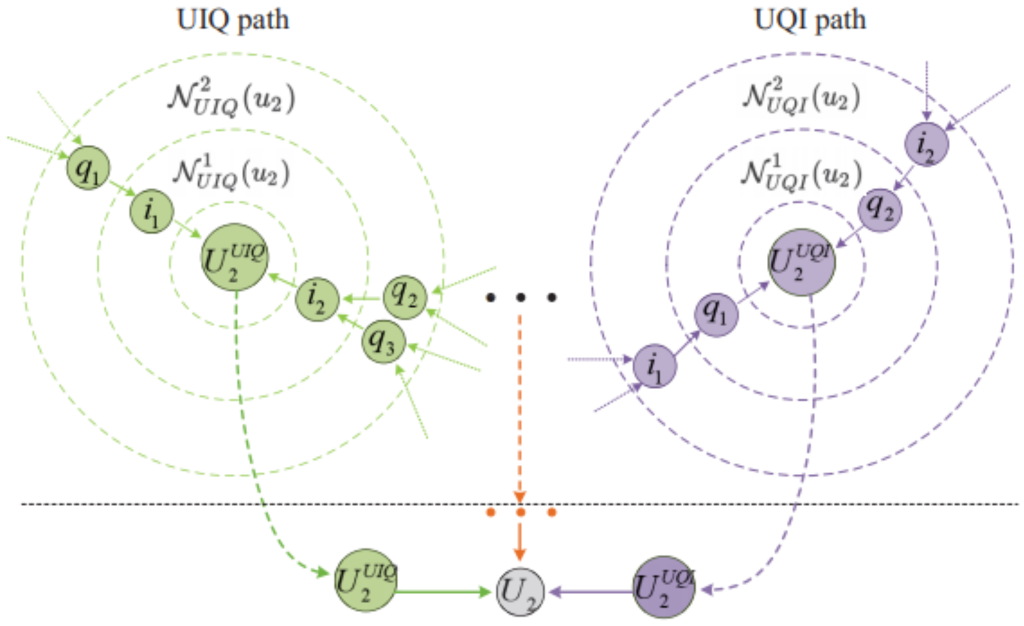
MEIRec: $O(d*N+n*h) \approx O(d*N)$. $N \ll M$

## New objects

- The embeddings of new objects (item, query) can be computed by the trained term embedding in the testing phase

# Metapath-guided Heterogeneous Graph Neural Network




- Initial embeddings.

$$E_{q_2} = g(e_{w_1}, e_{w_n}), E_{i_2} = g(e_{w_1}, e_{w_{n-1}}, e_{w_n}),$$

Trainable parameters

- Neighbor aggregation

$$I_j^{\mathrm{UIQ}} = g(E_{q_1}, E_{q_2}, \cdots),$$

$$U_i^{\mathrm{UIQ}} = g(I_1^{\mathrm{UIQ}}, I_2^{\mathrm{UIQ}}, \cdots),$$

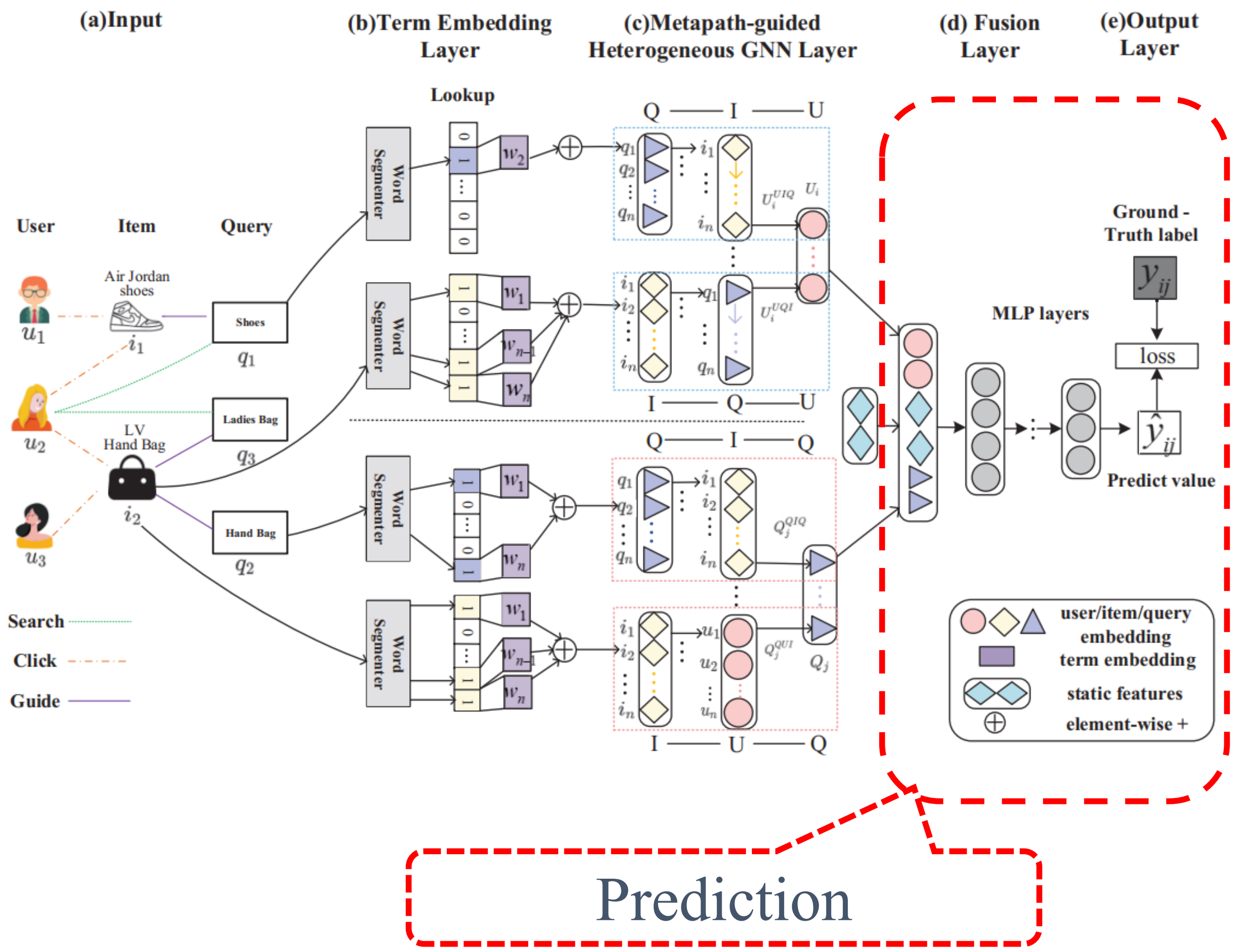
Different aggregation function

- Metapath Aggregating

Different aspect information

$$U_i = g(U_i^{\rho_1}, U_i^{\rho_2}, \cdots, U_i^{\rho_k}),$$

# Prediction




- Predict score

$$\hat{y}_{ij} = sigmoid(f(U_i \oplus Q_j \oplus S_{ij})),$$

- Point-wise loss function

$$J = \sum_{i,j \in \mathcal{Y} \cup \mathcal{Y}^-} (y_{ij} log \hat{y}_{ij} + (1 - y_{ij}) log(1 - \hat{y}_{ij})),$$

## Baselines

- NeuMF
- LR
- LR+DW/MP
- DNN
- DNN+DW/MP
- GBDT
- GBDT+DW/MP

## Metric

- Offline:AUC
- Online:CTR、Unique Click、UCTR

## Datasets

**Table 1: The statistics of the datasets.**

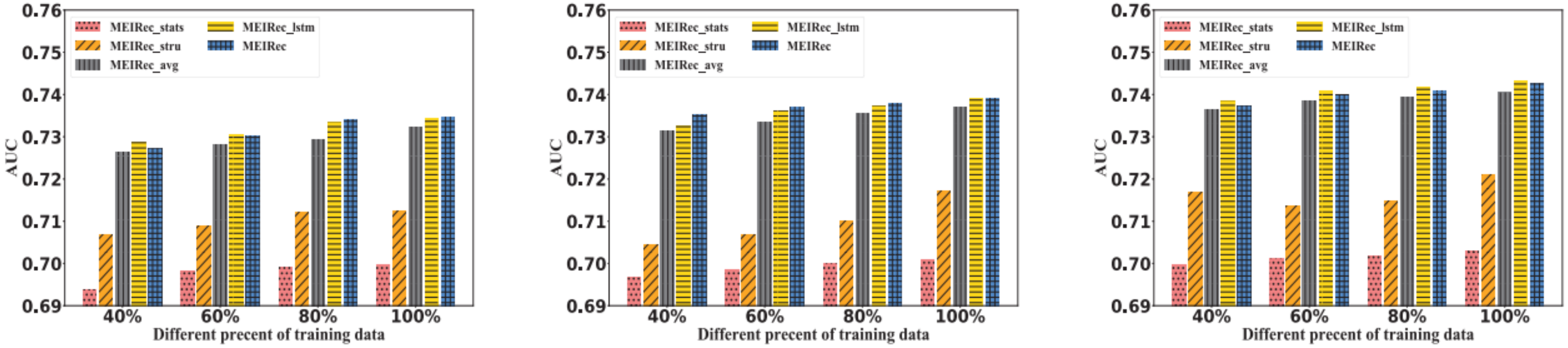
| Dataset | 1-day | 3-day | 5-day |
|---|---|---|---|
| Training size (positive) | 2,000,000 | 6,000,000 | 9,999,999 |
| Training size (all) | 8,000,000 | 23,999,998 | 39,999,997 |
| Validation size (positive) | 2,000,000 | 2,000,000 | 1,949,143 |
| Validation size (all) | 7,999,997 | 8,000,000 | 7,949,142 |
| Train users | 4,792,621 | 11,489,531 | 16,419,735 |
| Train queries | 871,133 | 1,653,865 | 2,163,574 |
| Validation users | 4,819,489 | 4,809,497 | 4,790,912 |
| Validation queries | 876,636 | 859,488 | 787,672 |
| New users in validation set | 3,666,692 | 2,613,695 | 2,064,564 |
| Density | $4.8 \times 10^{-7}$ | $3.1 \times 10^{-7}$ | $2.8 \times 10^{-7}$ |

# Experiments

| Method | 1-day | | | | 3-day | | | | 5-day | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 40% | 60% | 80% | 100% | 40% | 60% | 80% | 100% | 40% | 60% | 80% | 100% |
| NeuMF | 0.6014 | 0.6066 | 0.6136 | 0.6143 | 0.6168 | 0.6218 | 0.6249 | 0.6291 | 0.6172 | 0.6224 | 0.6246 | 0.6295 |
| LR | 0.6854 | 0.6838 | 0.6884 | 0.6889 | 0.6844 | 0.6863 | 0.6857 | 0.6865 | 0.6817 | 0.6831 | 0.6827 | 0.6836 |
| LR+DW | 0.6878 | 0.6904 | 0.6898 | 0.6930 | 0.6888 | 0.6896 | 0.6898 | 0.6900 | 0.6838 | 0.6842 | 0.6863 | 0.6867 |
| LR+MP | 0.6918 | 0.6936 | 0.6950 | 0.6969 | 0.6919 | 0.6930 | 0.6933 | 0.6933 | 0.6874 | 0.6890 | 0.6898 | 0.6899 |
| DNN | 0.6939 | 0.6981 | 0.6991 | 0.6997 | 0.6966 | 0.6985 | 0.6999 | 0.7008 | 0.6996 | 0.7011 | 0.7017 | 0.7029 |
| DNN+DW | 0.6962 | 0.6980 | 0.7003 | 0.7024 | 0.7005 | 0.7017 | 0.7024 | 0.7030 | 0.7017 | 0.7029 | 0.7040 | 0.7047 |
| DNN+MP | 0.6984 | 0.6992 | 0.7024 | 0.7057 | 0.7025 | 0.7040 | 0.7051 | 0.7057 | 0.7017 | 0.7044 | 0.7060 | 0.7069 |
| GBDT | 0.7071 | 0.7071 | 0.7067 | 0.7073 | 0.7070 | 0.7071 | 0.7072 | 0.7071 | 0.7067 | 0.7068 | 0.7072 | 0.7066 |
| GBDT+DW | 0.7114 | 0.7119 | 0.7112* | 0.7118* | 0.7109 | 0.7106 | 0.7106 | 0.7104 | 0.7109 | 0.7112 | 0.7109 | 0.7114 |
| GBDT+MP | 0.7122* | 0.7127* | 0.7110 | 0.7111 | 0.7123* | 0.7122* | 0.7122* | 0.7124* | 0.7118* | 0.7114* | 0.7114* | 0.7120* |
| MEIRec | **0.7273** | **0.7302** | **0.7339** | **0.7346** | **0.7352** | **0.7369** | **0.7380** | **0.7390** | **0.7372** | **0.7401** | **0.7409** | **0.7425** |
| Improvement | 2.1% | 2.5% | 3.2% | 3.2% | 3.2% | 3.5% | 3.6% | 3.7% | 3.6% | 4.0% | 4.1% | 4.3% |

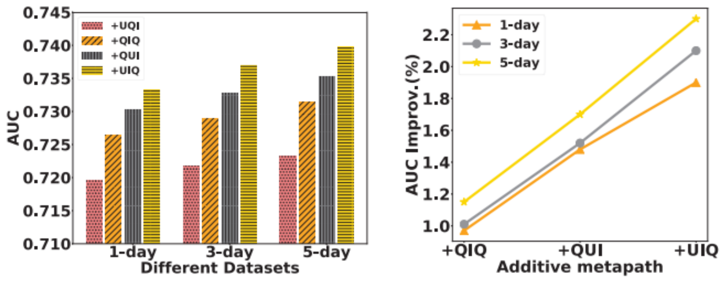
**MEIRec significantly outperforms GBDT, DNN, and MF based methods**

## MEIRec with different aggregation strategies



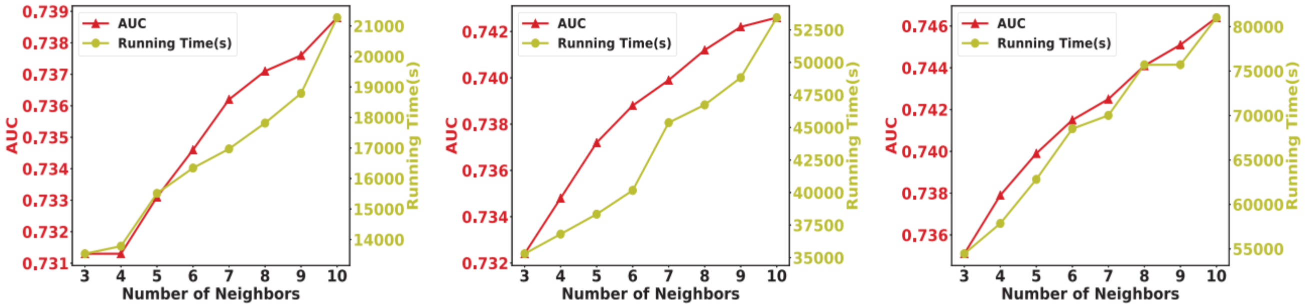

(a) 1-day (b) 3-day (c) 5-day

## MEIRec with additive metapaths




(a) The AUC performances of additive metapths. (b) The AUC improvements of additive metapths.

## MEIRec with different number of neighbors




(a) 1-day (b) 3-day (c) 5-day

Table 3: Online A/B testing experiments results.

| Data | Methods | CTR | Unique Click | UCTR |
|---|---|---|---|---|
| Android | GBDT | 1.746% | 256,116 | 13.939% |
| | MEIRec | 1.758% | 260,634 | 14.229% |
| | Improvement | 0.70% | 1.76% | 2.07% |
| IOS | GBDT | 0.7687% | 62,462 | 5.2579% |
| | MEIRec | 0.8056% | 65,895 | 5.5436% |
| | Improvement | 4.79% | 5.50% | 5.43% |
| Total | GBDT | 1.4035% | 318,578 | 10.5252% |
| | MEIRec | 1.4252% | 326,529 | 10.8052% |
| | Improvement | 1.54% | 2.50% | 2.66% |

**MEIRec significantly improves key metrics considered by the platform and attracts more new users to search the recommended query**

## Summary

- We study the intent recommendation problem which plays an important role in increasing user activity and stickiness  in mobile e-commerce
- We model objects and interactions in intent recommendation system with a HIN and propose a novel metapath-guided GNN method for intent recommendation
- The extensive results on offline and online experiments demonstrate the effectiveness of our proposed model

| Method | Node-level Aggregating | Semantic-level Aggregating | Task |
|---|---|---|---|
| MEIRec | Avg/CNN/RNN | Concatenation | Intent Recommendation |

# Spam Review Detection with Graph Convolutional Networks

Ao Li*
Alibaba Group
Hangzhou, China
jianzhen.la@alibaba-inc.com

Zhou Qin*
Alibaba Group
Hangzhou, China
qinzhou.qinzhou@alibaba-inc.com

Runshi Liu
Alibaba Group
Hangzhou, China
runshi.lrs@alibaba-inc.com

Yiqun Yang
Alibaba Group
Beijing, China
yiqun.yyq@alibaba-inc.com

Dong Li
Alibaba Group
Hangzhou, China
shiping@taobao.com

# Spam review detection at Xianyu



找兼职，加微信XXX
Seek part-time jobs, add XXX

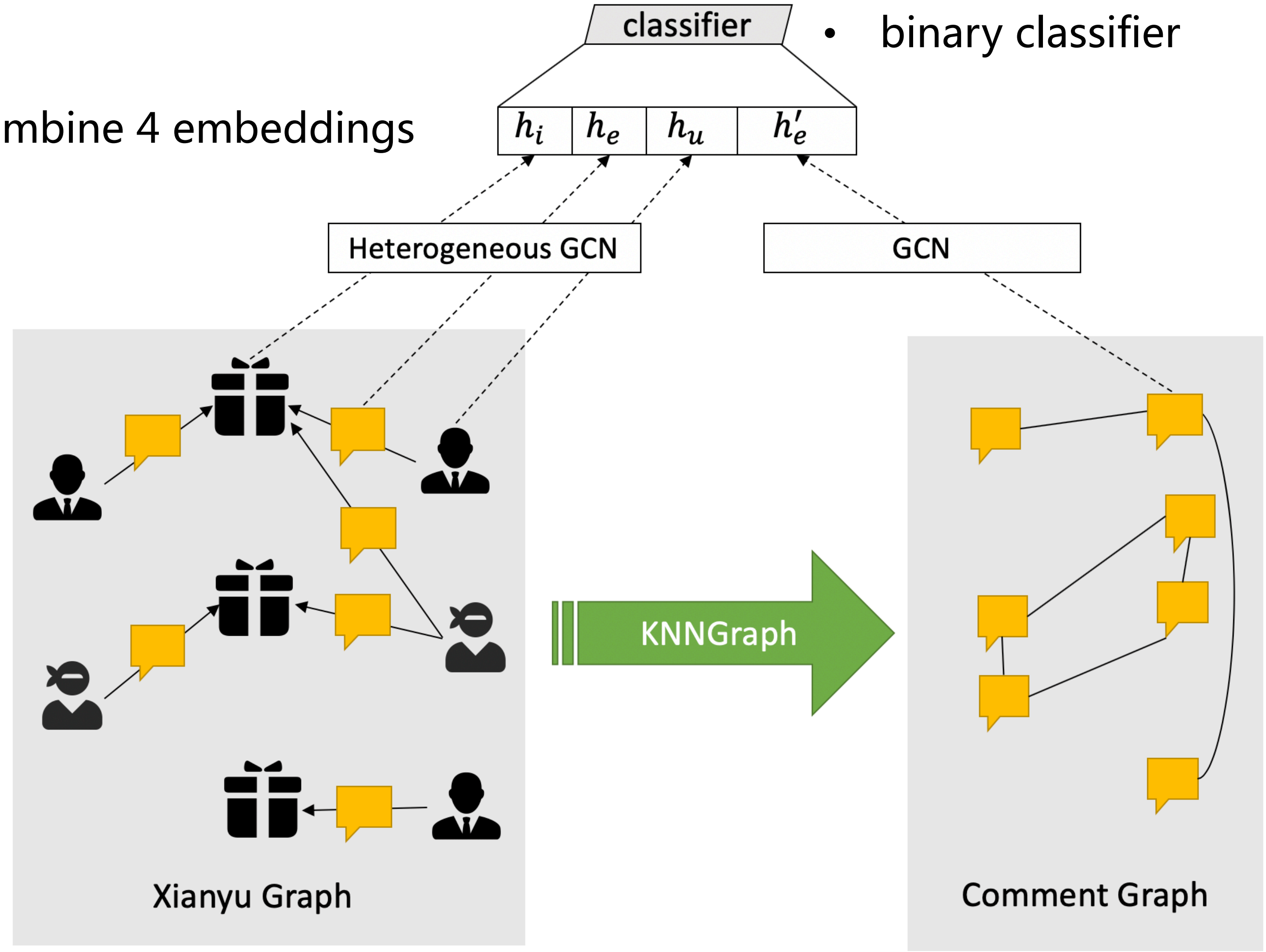
需要刷粉请联系我XXX
Contact me for water armies XXX

主营国烟送礼佳品，茄xxxx
Great tobaccos, add XXX

减肥药,看昵称联系
weight loss pills, see nick name for details

## Spam review detection at Xianyu

- binary classifier
- combine 4 embeddings




Heterogeneous graph

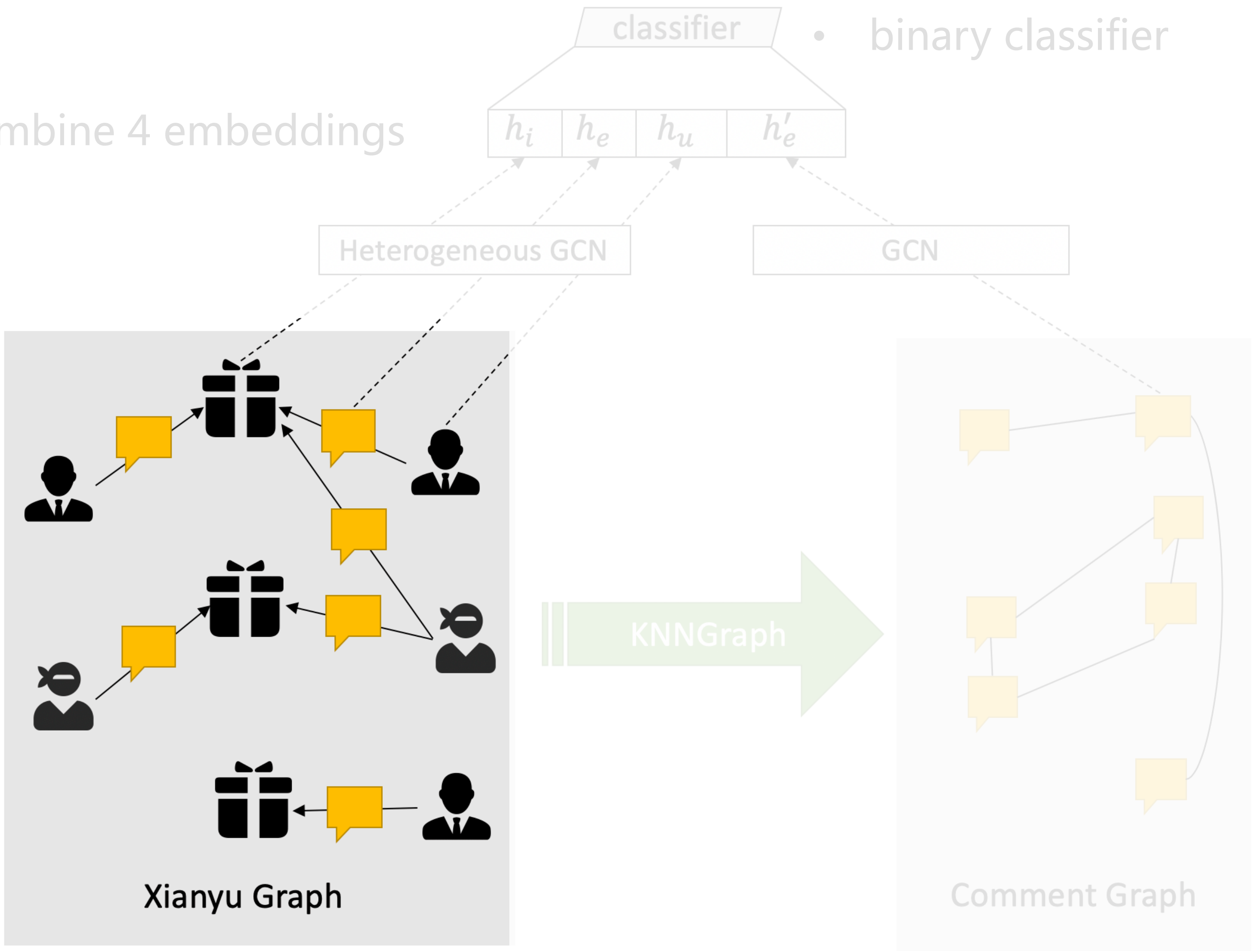
- item-comment-user
- metapaths
- 2-hop neighborhood aggregation

homogeneous graph

- comment-comment
- made by KNNGraph
- smoothing process

## Heterogeneous GNN for Spam Review Detection

# Main framework of our Algorithm



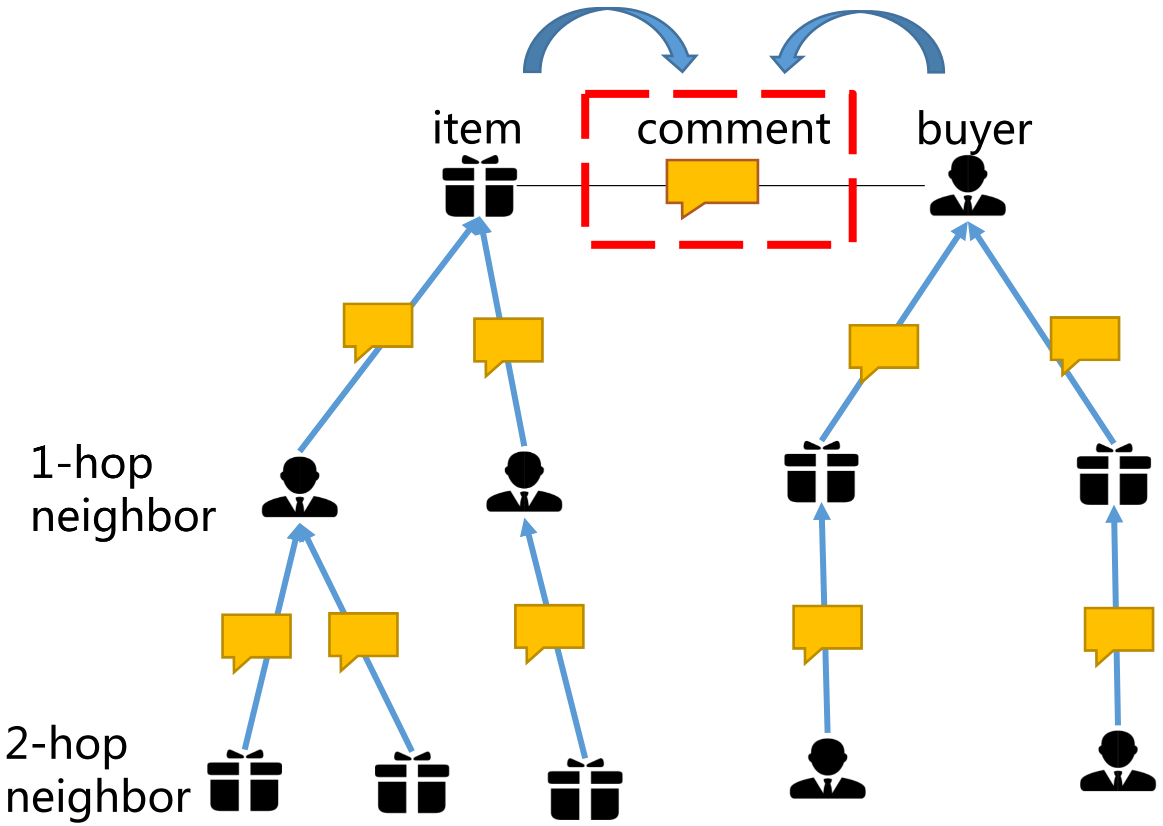

### Heterogeneous graph

- item-comment-user
- metapaths
- 2-hop neighborhood aggregation

### homogeneous graph

- comment-comment
- made by KNNGraph
- smoothing process

# Metapaths and GCN algorithm on heterogenous graph




How to make the model more robust?

Based on three embeddings!

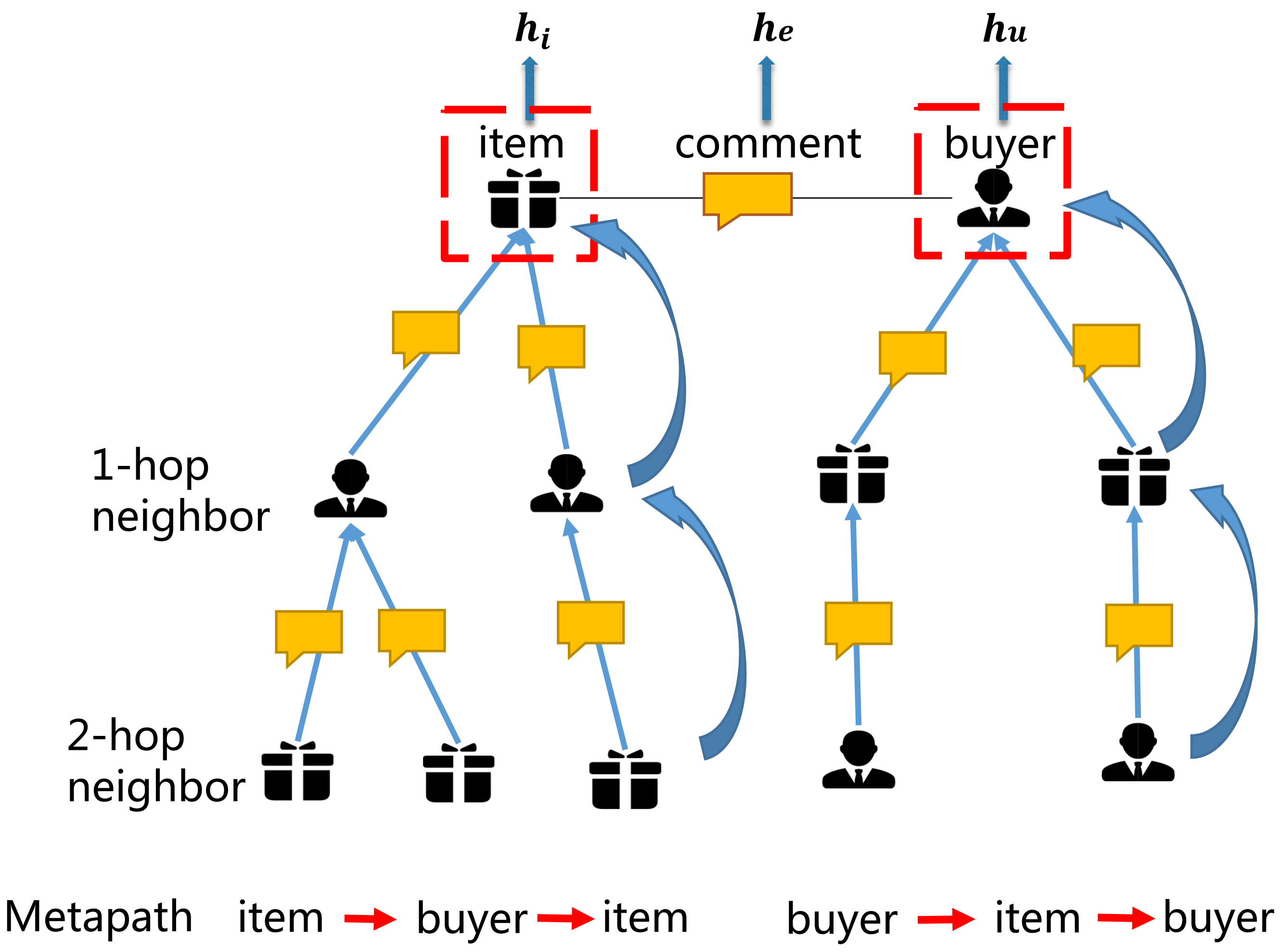
- comment's embedding
- item's embedding which receive the comment
- buyer's embedding who post the comment

Adversarial text evaluation by spammers

加微信XXX → 茄微信 XXX → 茄薇XXX → +V XXX

add Wechat XXX → abd Wechat XXX → abd We XXX → +V XXX

# Metapaths and GCN algorithm on heterogenous graph (cont'd)




How to define item/buyer embeddings?

Through 2 Metapaths!

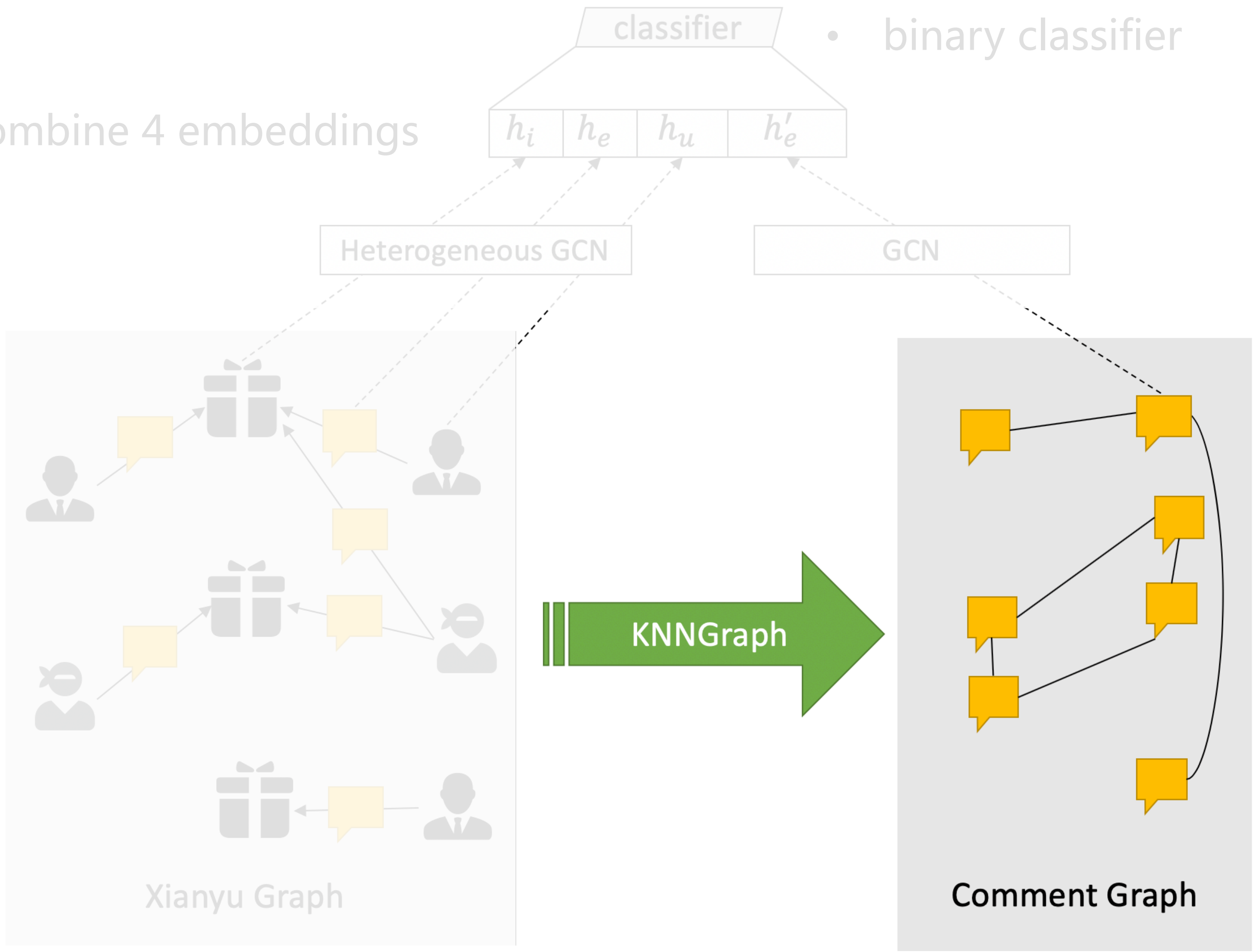
- Item's embedding is a fusion of it's features and features of related buyers
- Buyer's embedding is a fusion of his features and features of related items

# Main framework of our Algorithm

- binary classifier
- combine 4 embeddings

Heterogeneous graph

- item-comment-user
- metapaths
- 2-hop neighborhood aggregation

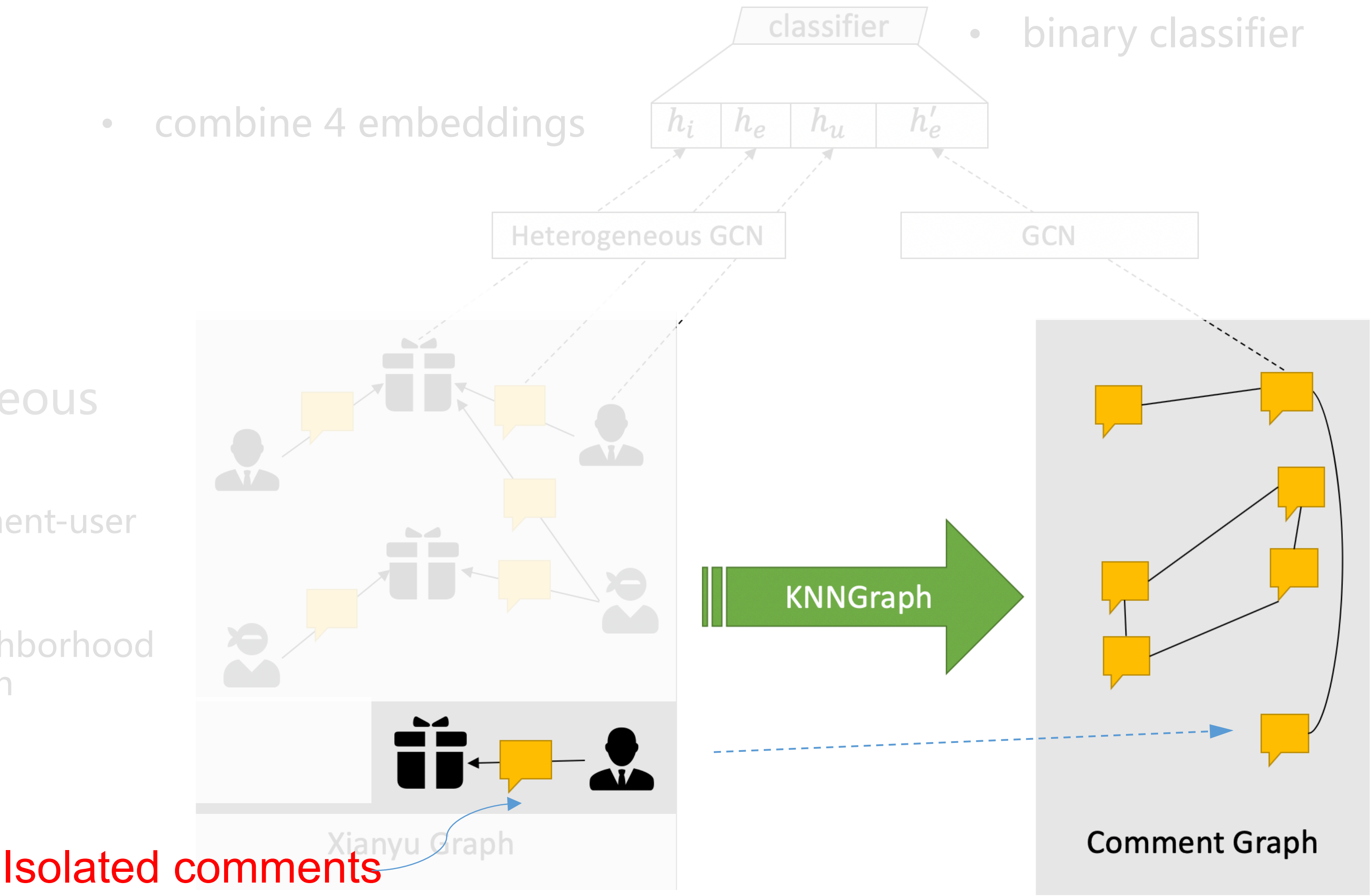



homogeneous graph

- comment-comment
- made by KNNGraph
- smoothing process

## Heterogeneous GNN for Spam Review Detection

# Main framework of our Algorithm




- binary classifier
- combine 4 embeddings

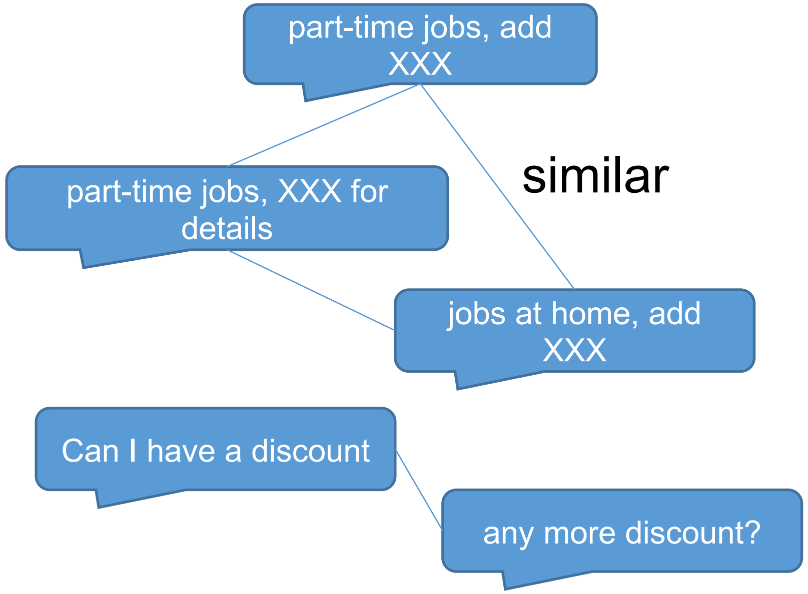
Heterogeneous graph

- item-comment-user
- metapaths
- 2-hop neighborhood aggregation

homogeneous graph

- comment-comment
- made by KNNGraph
- smoothing process

# GCN on comment graph




Why?

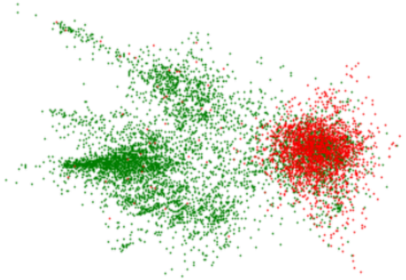
- Import more global context for isolated nodes

How?

- Group similar comments together through KNNGraph algorithm

- GCN model is actually a special form of Laplacian smoothing, which further helps alleviate the impact of adversarial text

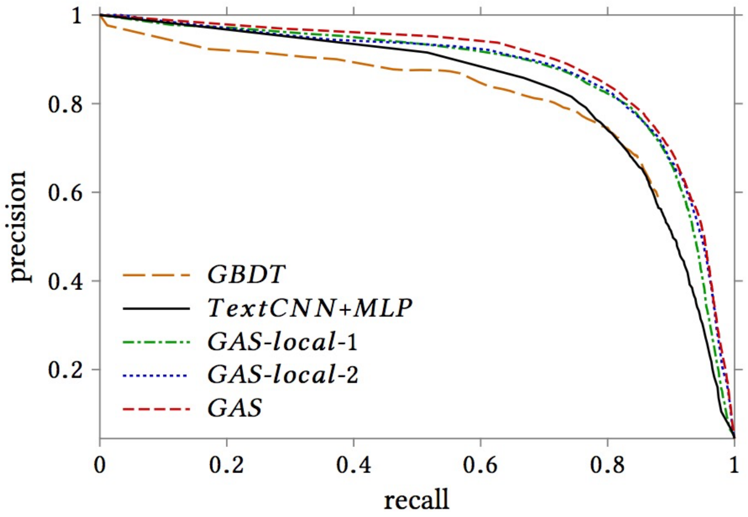
Without GCN smoothing

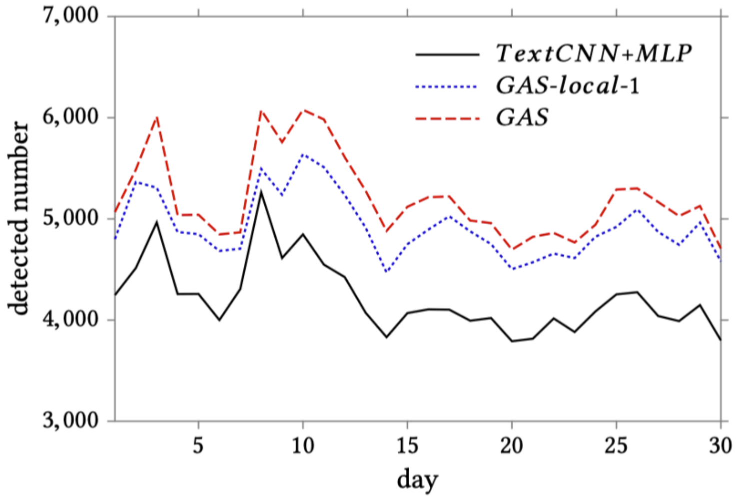


After GCN smoothing

# Heterogeneous GNN for Spam Review Detection

## Experiments




- GAS-local-1 vs. TextCNN+MLP
  - our model vs former deployed method (baseline), 16% lift
- GAS vs. GAS-local-2
  - our model vs 2-hop heterogenous graph 4% lift

| | method | AUC | F1 score | recall@90% precision |
|---|---|---|---|---|
| baseline | GBDT | 0.9649 | 0.7686 | 50.55% |
| | TextCNN+MLP | 0.9750 | 0.7784 | 54.86% |
| Xianyu graph only | GAS-local-1 | 0.9806 | 0.8138 | 66.90% |
| | GAS-local-2 | 0.9860 | 0.8143 | 67.02% |
| Comment with Xianyu graph | GAS | 0.9872 | 0.8217 | 71.02% |

**Table 2: Result comparison of offline experiments in terms of *AUC*, *F1-score* and *recall at 90% precision* which is denoted as *recall@90% precision*.**

CONTENTS

## Heterogeneous GNNs usually follow two steps:

- **Node-level Aggregating**: Aggregate neighbors via single meta-path.
- **Semantic-level Aggregating**: Aggregate rich semantics via multiple meta-paths.

(Note that heterogeneous edge can be view as a meta-path (length is 1).)

| Method | Node-level Aggregating | Semantic-level Aggregating | Task |
|---|---|---|---|
| HAN | Attention | Attention | Node classification/Clustering |
| HPN | Avg | Attention | Node classification/Clustering |
| HGAT | Attention | Attention | Short texts classification |
| MEIRec | Avg/CNN/RNN | Concatenation | Intent Recommendation |
| GAS | Attention | / | Spam Review Detection |

# 4 Conclusions

- Graph-structured data especially heterogeneous graph are ubiquitous.
- Heterogeneous graph neural network is a powerful graph representation technique based on deep learning.
- Heterogeneous graph and Heterogeneous GNN can be applied to diverse tasks (e.g., NLP and Recommendation).

[1]. Chuan Shi, Yitong Li, Jiawei Zhang, Yizhou Sun, Philip S. Yu. A survey on Heterogeneous Information Network Analysis. IEEE Transactions on Knowledge and Data Engineering, 29(1), 17-37, 2017 .
[2]. Chuan Shi, Philip S. Yu. Heterogeneous Information Network Analysis and Applications. Springer,2017
[3]. Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, PietroLiò, and Yoshua Bengio. 2018. Graph Attention Networks. In ICLR.
[4]. Wang, Xiao, Houye Ji, Chuan Shi, Bai Wang, Yanfang Ye, Peng Cui, and Philip S. Yu. Heterogeneous Graph Attention Network. In *The World Wide Web Conference*, pp. 2022-2032. ACM, 2019.
[5]. Linmei, Hu, Tianchi Yang, Chuan Shi, Houye Ji, and Xiaoli Li. Heterogeneous graph attention networks for semi-supervised short text classification. *EMNLP-IJCNLP*, pp. 4823-4832. 2019.
[6]. Shaohua Fan, Junxiong Zhu, Xiaotian Han, Chuan Shi, Linmei Hu, Biyu Ma, Yongliang Li. Metapath-guided Heterogeneous Graph Neural Network for Intent Recommendation. KDD 2019.

# Thank you !
# Q&A

More materials:
http://shichuan.org
https://zhuanlan.zhihu.com/p/95933043