



中南大學
CENTRAL SOUTH UNIVERSITY

文本图上的多模态表示学习 方法研究

答 辯 人：计科1901 庄纹纹
指导老师：王 森 章

2023/5/27





Contents

目录

01 研究背景

02 研究现状

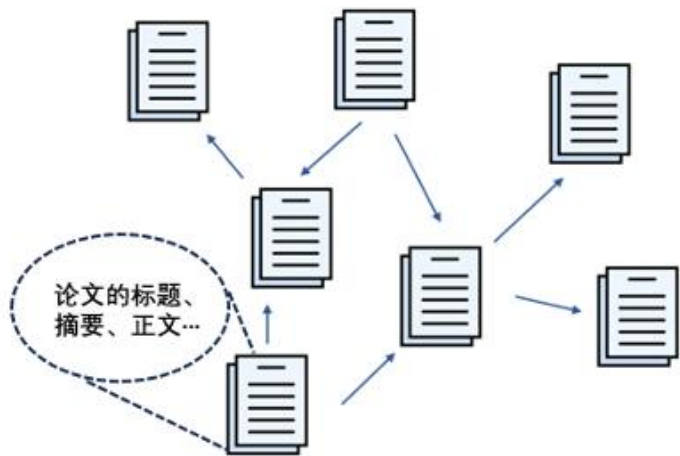
03 模型方法

04 实验结果

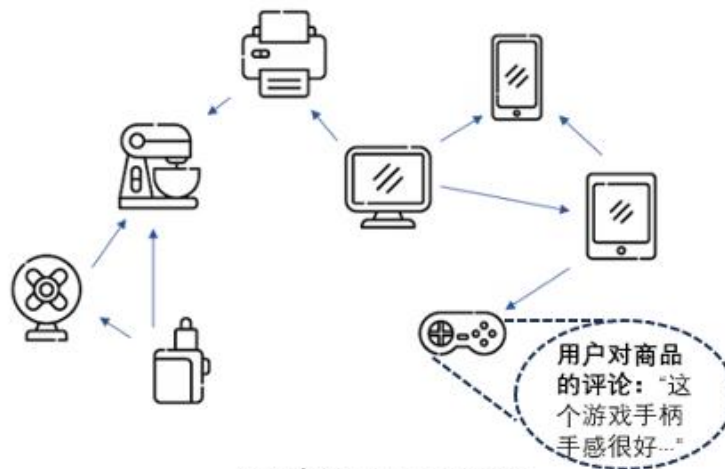
05 总结展望



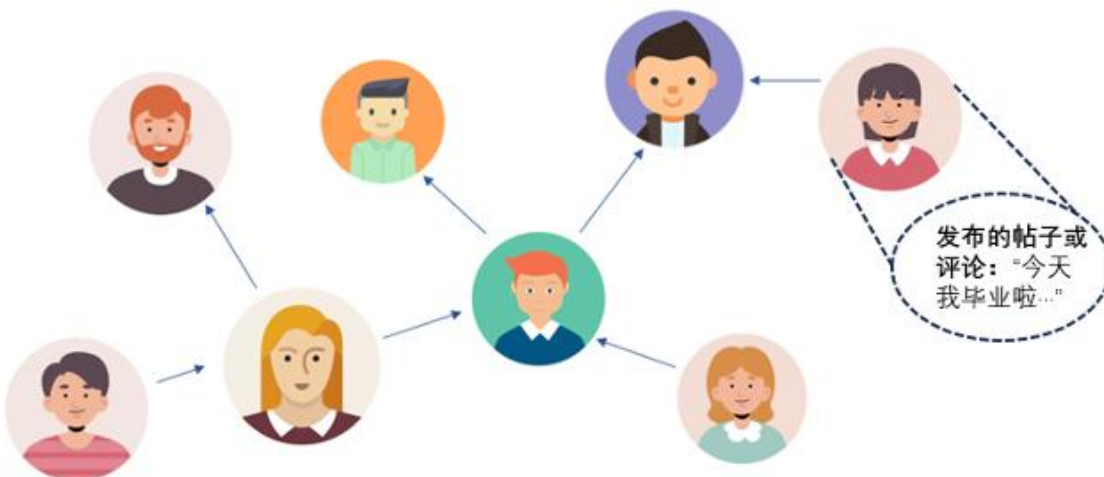
■ 文本图及其表示学习



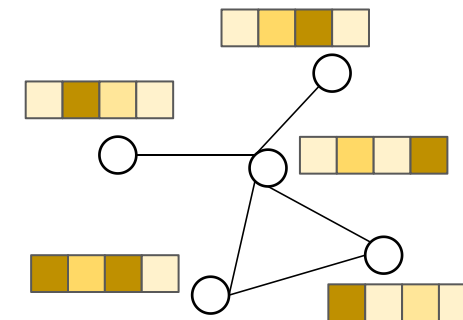
(a) 论文引用图



(b) 电商产品关系图



(c) 社交网络图



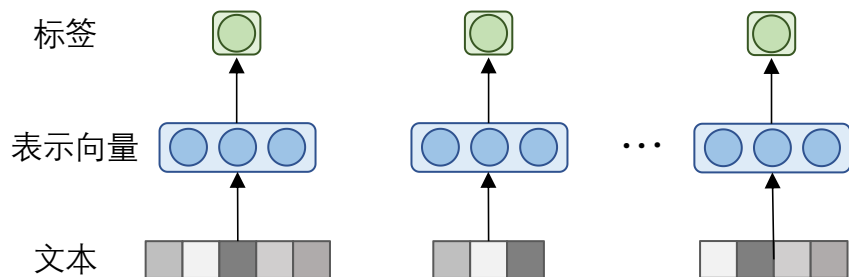
表示学习



下游任务

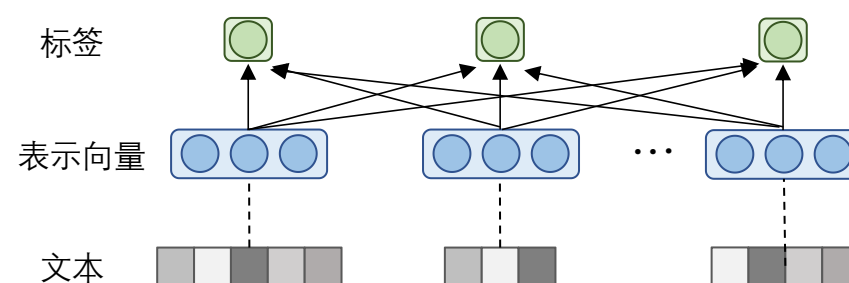
- 节点分类: 论文主题、用户特性等
- 链路预测: 用户之间的关系等

■ 文本图表示学习方法



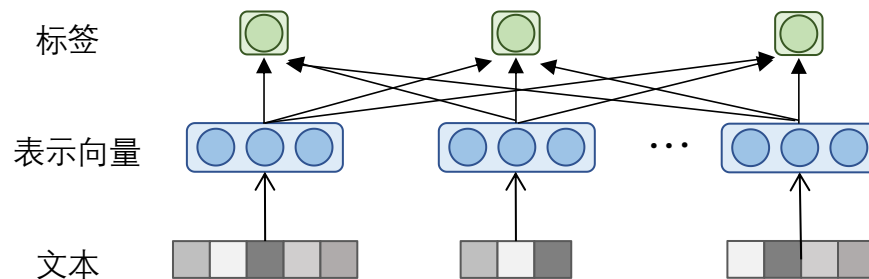
基于预训练语言模型的算法

忽略了节点之间的拓扑结构信息



基于图神经网络的算法

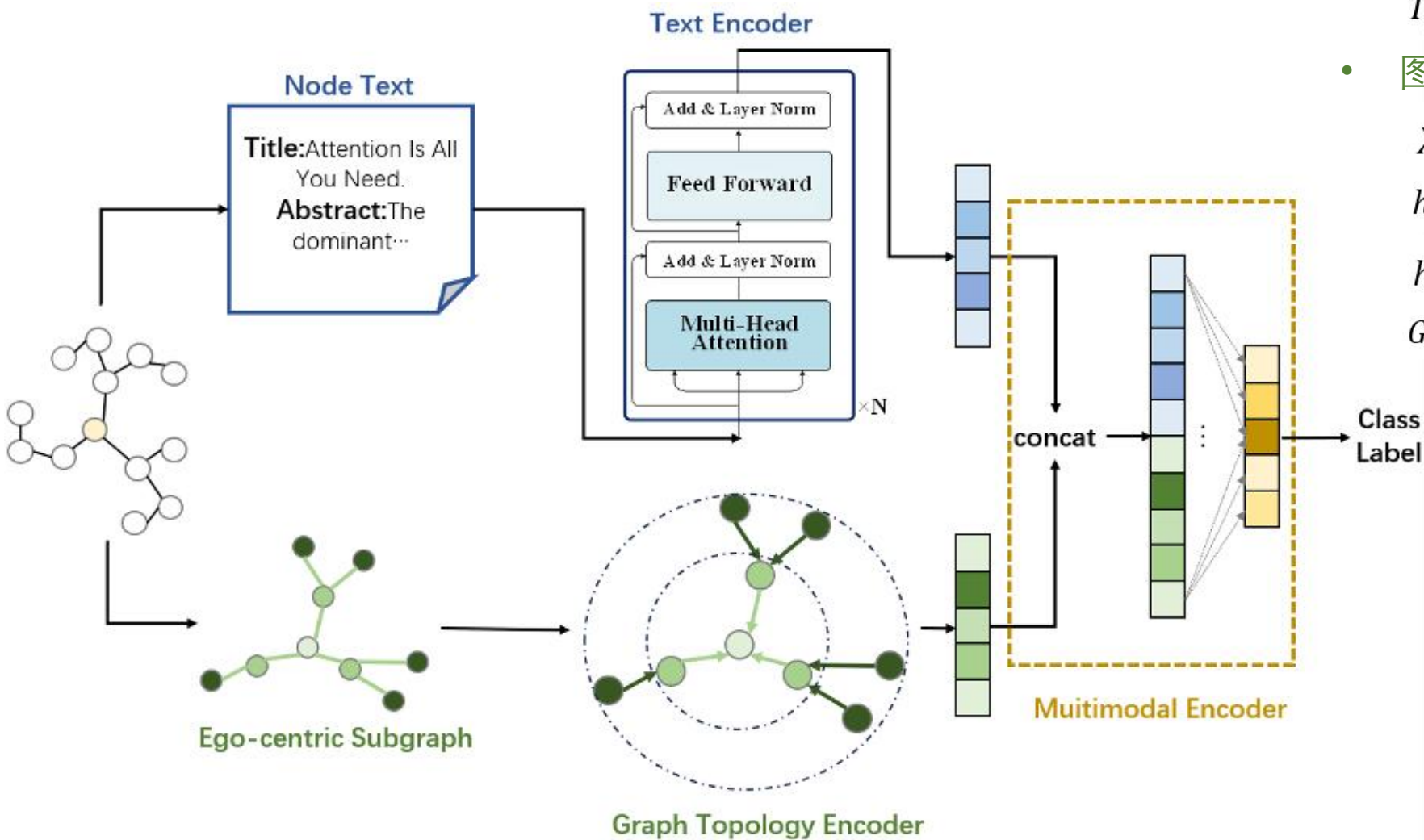
限制了模型对文本数据的深层次理解



基于级联架构的算法

可伸缩性差

■ 基于多模态的文本图表示学习算法



- 文本编码器

$$T_n = BERT(S_n)[hidden_states][[-1][0]]$$

- 图拓扑编码器

$$X = DeepWalk(V, A)$$

$$h'_i = AGGREGATE(\{x_j\}, j \in N(i))$$

$$h_i = \sigma(W \cdot CONCATENATE(h_i, h'_i))$$

$$G_n = GNN(NeighborSampler(X, A))$$

- 多模态编码器

$$H_n = W(CONCAT(T_n, G_n)) + b$$

- 损失函数

$$\hat{y} = \frac{e^{H_{ni}}}{\sum_{c=1}^C e^{H_{nc}}}$$

$$L = - \sum_{c=1}^C y_c \log(\hat{y}_c)$$



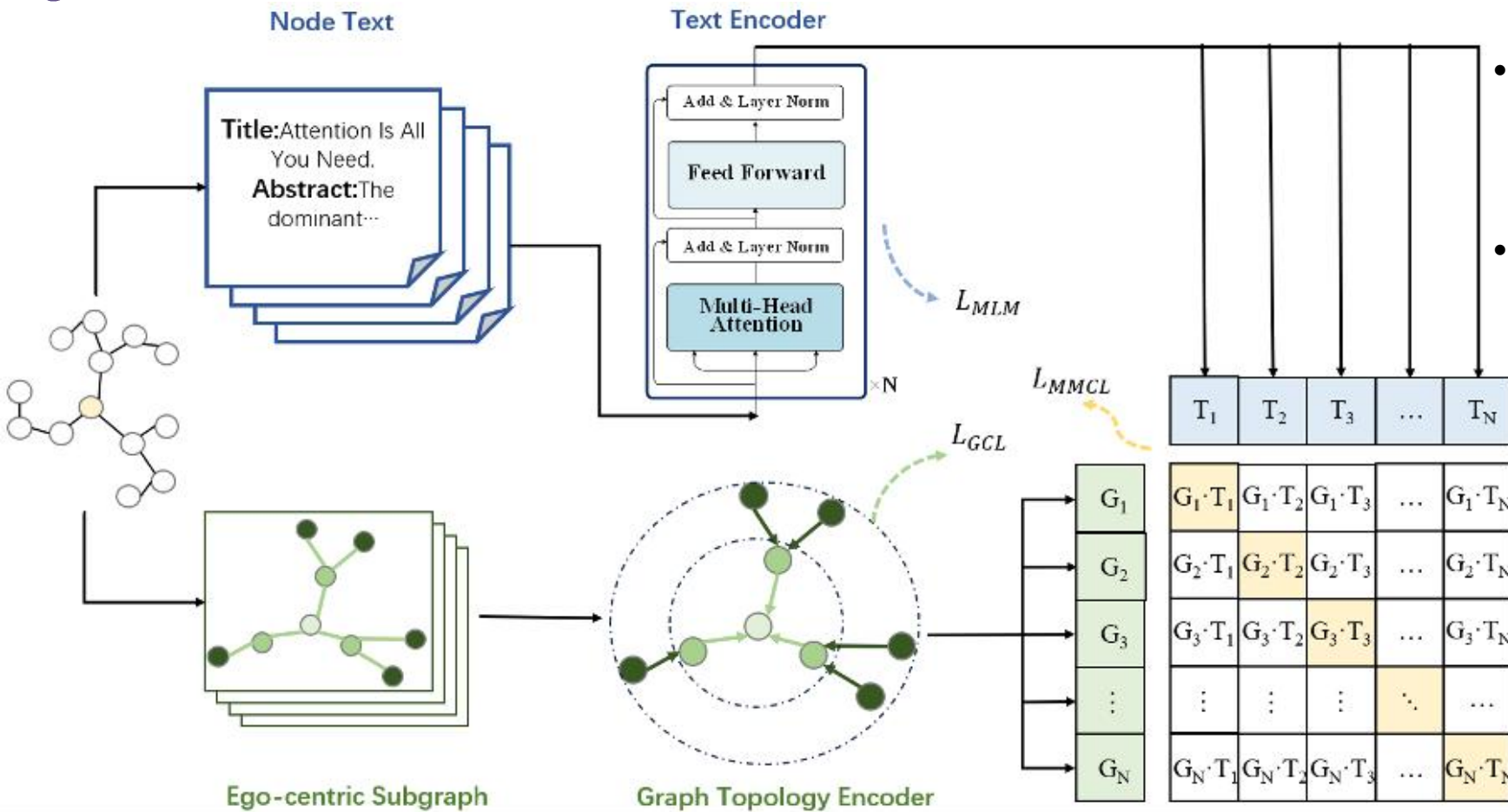
■ 基于多模态的文本图表示学习算法

Algorithm 1: MMGL

Input: Graph $G = (V, A, s_V)$. Model f_θ .

Output: Model f_θ .

```
1 begin
2    $X = \text{DeepWalk}(V, A)$ 
3   for each node  $i \in V$  do
4      $T_i = \text{BERT}(s_i)$ ; //  $s_i$  is the text of node  $i$ 
5      $G_i = \text{GNN}(\text{NeighborSampler}(A, X, i))$ ;
6      $H_i = W \cdot \text{Concat}(T_i, G_i) + b$ ;
7      $\hat{y} = \text{softmax}(H_i)$ ;
8      $L(f_\theta) = \text{CrossEntropy}(\hat{y}, y)$ ;
9      $f_\theta = f_\theta - \alpha \cdot \nabla L(f_\theta)$ ;
10  Return  $f_\theta$ ;
```



- 单模态预训练目标

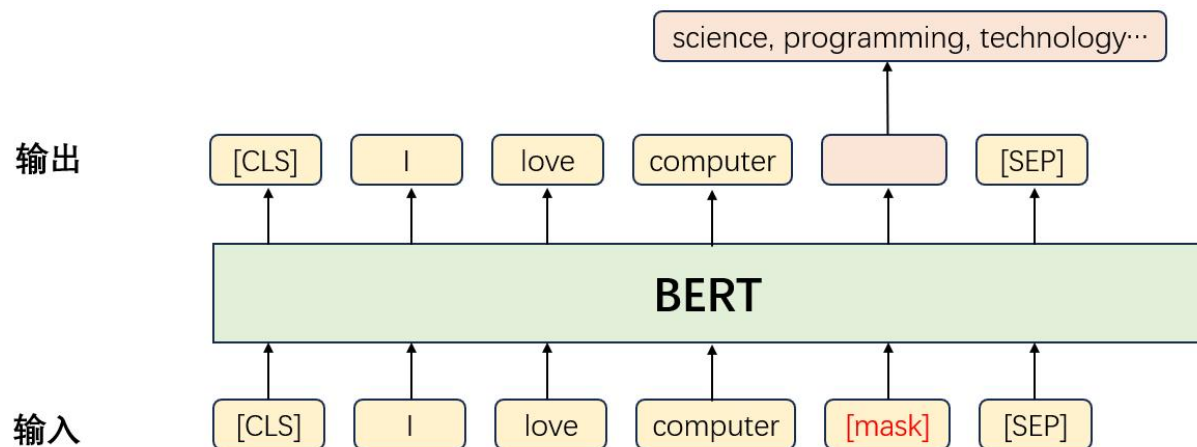
- 针对文本编码器：掩码语言建模
- 针对图拓扑编码器：图对比损失
- 多模态预训练目标
 - 多模态对比损失

- 多模态预训练目标

- 多模态对比损失 L_{MMCL}

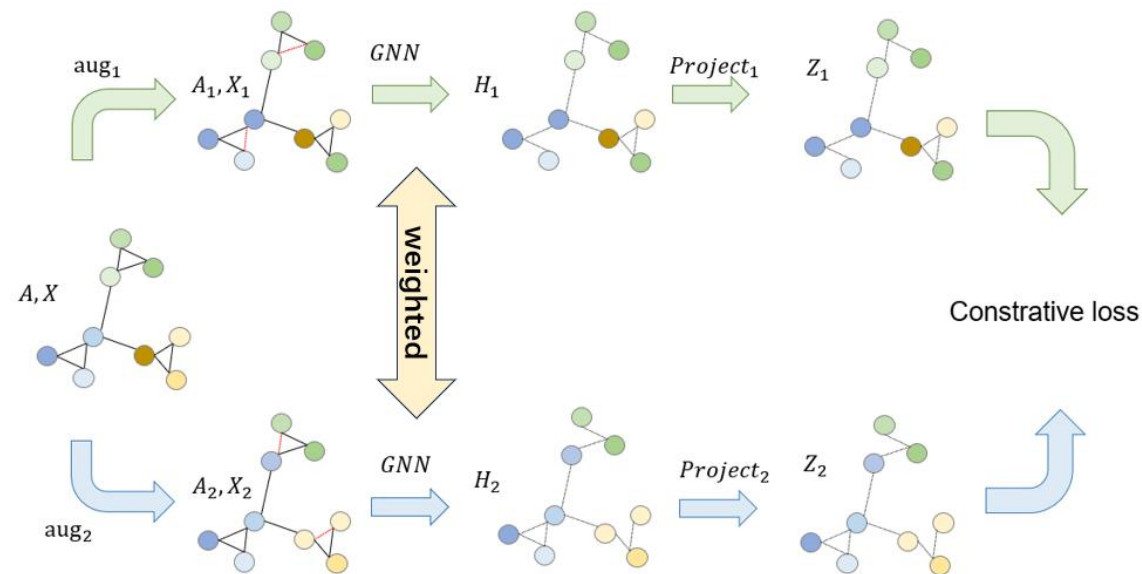
$$L_{MMCL} = -\log \frac{\exp(\text{sim}(T'_i, G'_i)/\tau)}{\sum_{k=1, k \neq i}^N \exp(\text{sim}(T'_i, G'_k)/\tau)}$$

- 单模态预训练目标
 - 掩码语言建模 L_{MLM}



$$L_{MLM} = -\frac{1}{|M_s|} \sum_{i \in M_s} \log P(s_i / s \setminus M_s)$$

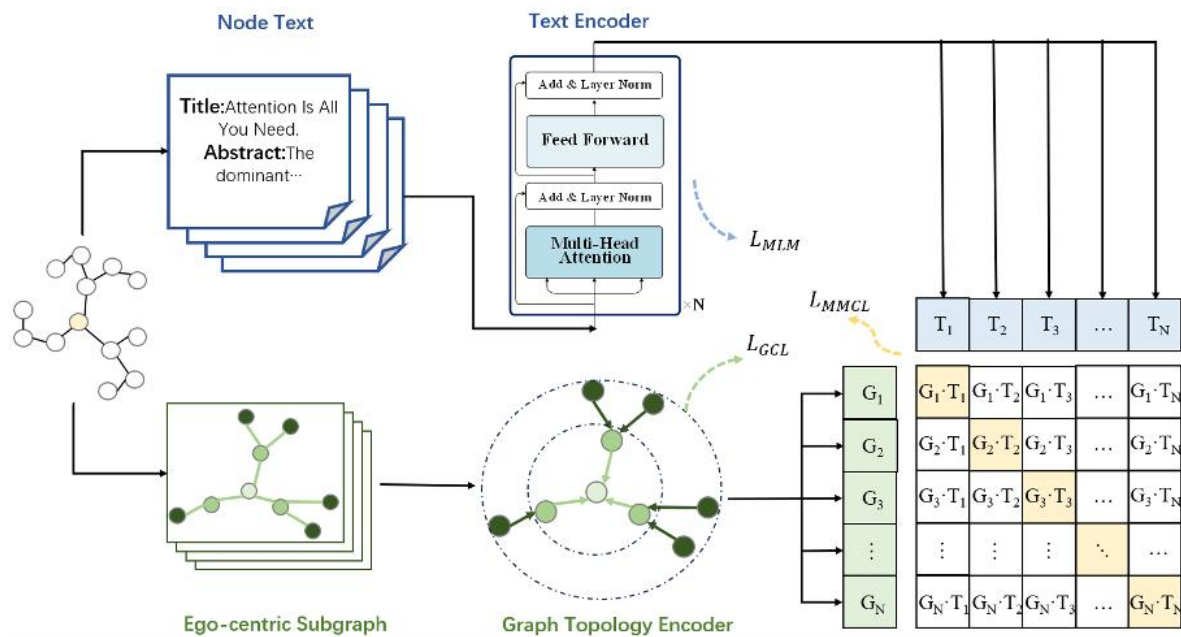
- 图对比损失 L_{GCL}



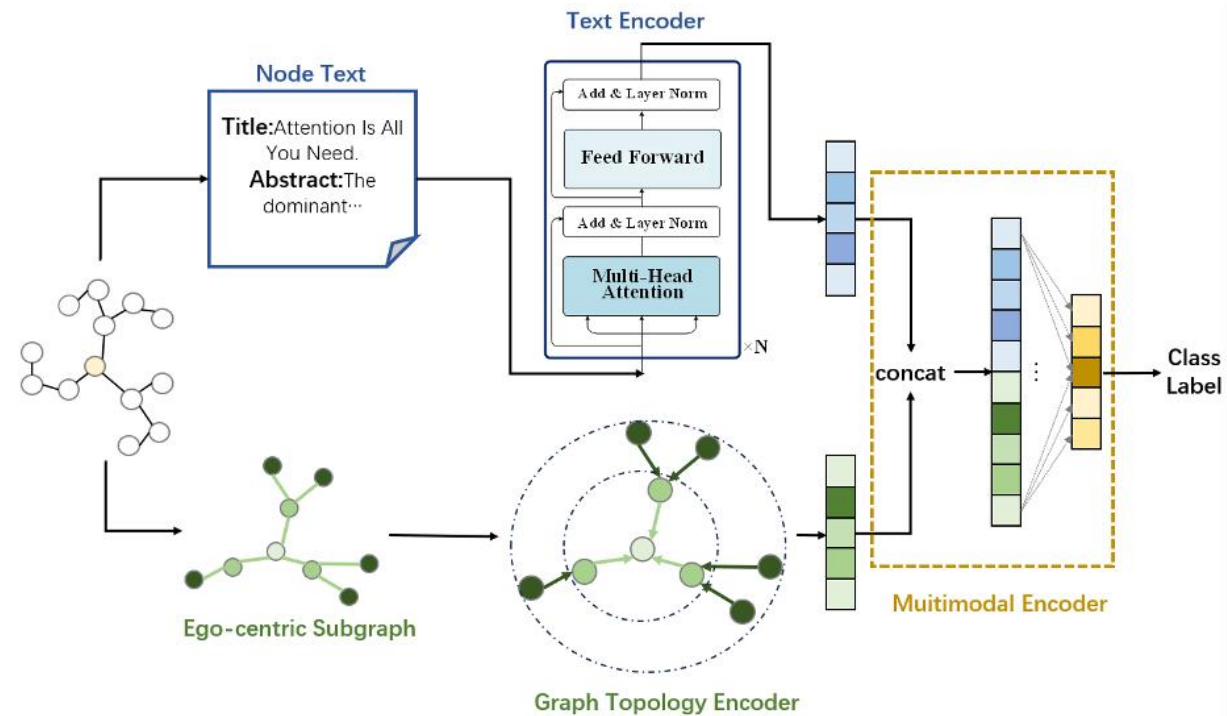
$$L_{GCL} = -\log \frac{\exp(\text{sim}(Z_{1i}, Z_{2i})/\tau)}{\sum_{k=1, k \neq i}^N \exp(\text{sim}(Z_{1i}, Z_{2k})/\tau)}$$

■ 基于多模态的文本图表示学习算法

• 预训练



• 微调





数据集介绍

- ogbn-arxiv
 - 是一个大规模的Arxiv论文引用图
 - 节点的文本属性：该论文的标题和摘要
 - 边：论文之间的引用关系
 - 标签：论文的主题类别
- Amazon-Electronics
 - 电商平台上的电子产品评论数据集
 - 节点的文本属性：用户对该商品的评论
 - 边：商品之间的共同购买关系
 - 标签：商品的类别

表 4-1 ogbn-arxiv 数据统计信息

	信息	数量	百分比
基本信息	节点数	169343	\
	边数	1166243	\
	节点平均度数	13.7	\
	标签类别数	40	\
数据集划分	训练集	90941	53.70%
	验证集	29799	17.60%
	测试集	48603	28.70%

表 4-2 Amazon-Electronics 数据统计信息

	信息	数量	百分比
基本信息	节点数	48362	\
	边数	500939	\
	节点平均度数	18.07	\
	标签类别数	12	\
数据集划分	训练集	18722	38.71%
	验证集	7419	15.34%
	测试集	22221	45.95%

评价指标

- 准确率

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$



■ 准确性

表 4-3 各算法在 ogbn-arxiv 数据集上的实验结果

		Val. accuracy	Test accuracy
GNN (DW Emb.)	GCN	0.7083	0.6975
	GraphSage	0.7202	0.7072
	GAT	0.7158	0.708
GNN (LM Emb.)	GCN	0.7178	0.705
	GraphSage	0.736	0.7195
	GAT	0.725	0.7096
LM-ft	BERT-tiny	0.7205	0.7016
	Deberta-Base	0.7385	0.7226
Cascaded TG	BERT-tiny	0.7398	0.7285
MMGL(ours)	BERT-tiny	<u>0.7549</u>	<u>0.7409</u>
	Deberta-Base	0.7623	0.7573

- 1) 比较GNN(DW Emb.)和GNN(LM Emb.): 文本信息对于文本图表示学习的重要性
- 2) 比较GNN中的GCN、GraphSage和GAT: GAT和GraphSAGE的效果优于GCN
- 3) 比较LM-ft中的BERT-tiny和Deberta-Base: 使用规模更大的语言模型可以带来更强大的编码能力
- 4) 比较Cascaded TG和MMGL: 我们的模型对显存的要求更小, 不会被迫牺牲模型的性能, 实验结果更优



■ 准确性

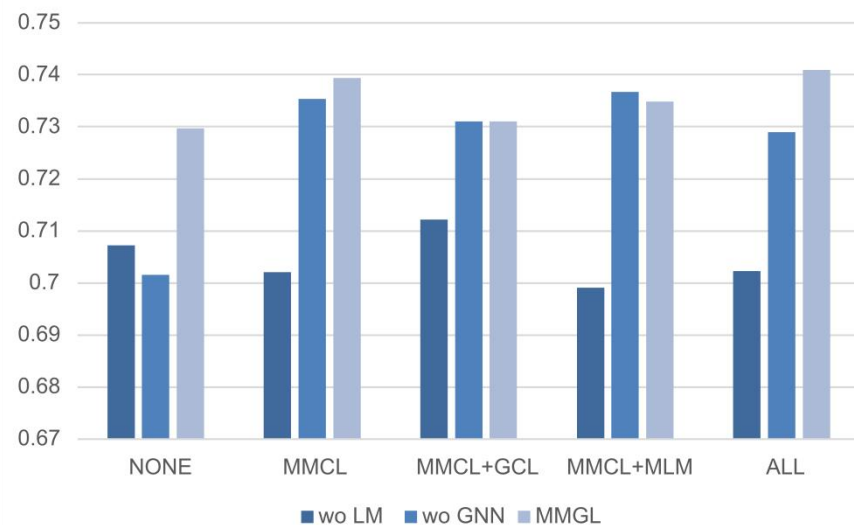
表 4-4 各算法在 Amazon-Electronics 数据集上的实验结果

		Val. accuracy	Test accuracy
GNN (DW Emb.)	GCN	0.8184	0.8024
	GraphSage	0.8358	0.8194
	GAT	0.8389	0.8214
GNN (LM Emb.)	GCN	0.8296	0.8201
	GraphSage	0.8467	0.8325
	GAT	0.8445	0.833
PLM	BERT-tiny	0.7027	0.6697
	Deberta-Base	0.7372	0.7046
Cascaded TG	BERT-tiny	0.8354	0.819
MMGL(ours)	BERT-tiny	<u>0.8473</u>	<u>0.8384</u>
	Deberta-Base	0.8485	0.8391

■ 消融实验

表 4-5 ogbn-arxiv 数据集消融实验结果

Pretrain Strategy	Model	BERT-tiny		Deberta-Base	
		Val. acc.	Test acc.	Val acc.	Test acc.
NONE	wo LM	0.7202	0.7072	\	\
	wo GNN	0.7205	0.7016	0.7385	0.7226
	MMGL	0.7408	0.7297	0.7457	0.7297
MMCL	wo LM	0.7151	0.7021	0.7158	0.7003
	wo GNN	0.746	0.7354	0.7546	0.7442
	MMGL	0.752	0.7394	0.7544	0.7442
MMCL+GCL	wo LM	0.7161	0.7122	0.7141	0.7092
	wo GNN	0.7457	0.731	0.7561	0.7488
	MMGL	0.7527	0.731	0.7551	0.7438
MMCL+MLM	wo LM	0.714	0.6991	0.7158	0.7028
	wo GNN	0.7471	0.7367	0.7601	0.7501
	MMGL	0.7525	0.7348	0.7609	0.7561
ALL	wo LM	0.7165	0.7023	0.714	0.7046
	wo GNN	0.7475	0.729	0.7604	0.7525
	MMGL	0.7549	0.7409	0.7623	0.7573

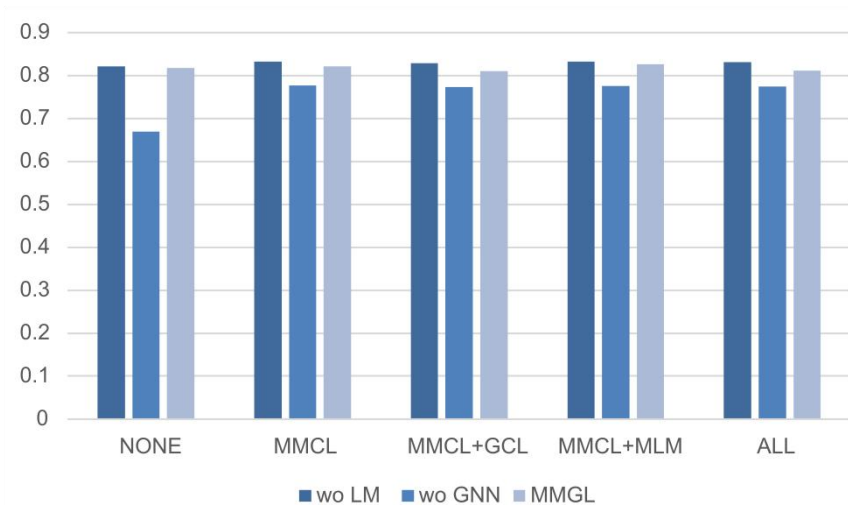


- 1) 分析模型内不同模块对模型性能的影响
- 2) 分析文本编码器参数量对模型性能的影响
- 3) 分析预训练策略的选择对模型性能的影响
- 4) 分析消融实验中两个数据集表现的差异分析得出模型在不同特征的数据上的效果

■ 消融实验

表 4-6 Amazon-Electronics 数据集消融实验结果

Pretrain Strategy	Model	BERT-tiny		Deberta-Base	
		Val. acc.	Test acc.	Val acc.	Test acc.
NONE	wo LM	0.8389	0.8214	\	\
	wo GNN	0.7027	0.6697	0.7372	0.7046
	MMGL	0.8304	0.8173	0.8217	0.8073
MMCL	wo LM	0.8435	0.8329	0.8399	0.8275
	wo GNN	0.802	0.7776	0.8133	0.7839
	MMGL	0.8354	0.8209	0.8245	0.8066
MMCL+GCL	wo LM	0.8431	0.8291	0.8405	0.8319
	wo GNN	0.7977	0.7735	0.8083	0.7787
	MMGL	0.825	0.8099	0.8054	0.7807
MMCL+MLM	wo LM	0.8426	0.8321	0.8397	0.8229
	wo GNN	0.8006	0.7762	0.8201	0.7922
	MMGL	0.8428	0.8267	0.8259	0.8043
ALL	wo LM	0.8435	0.8318	0.8403	0.8288
	wo GNN	0.7981	0.7748	0.8098	0.7761
	MMGL	0.831	0.8121	0.8114	0.7828





05

总结展望





■ 总结

- 将图的拓扑结构信息看成一种模态，提出了一种基于多模态的文本图表示学习算法
- 采用了多模态和单模态的预训练目标
- 在两种类型的数据集上取得的性能优于基准方法，证明我们方法的有效性

■ 展望

- 增加考虑其他模态信息，如图像、音频等
- 考虑更多下游任务，如多模态图数据上设计链路预测等更多下游任务
- 考虑扩展我们的方法在其他领域和任务中的应用，如社交网络分析、推荐系统、生物信息学等领域

■ 致谢

- 感谢中南大学王森章教授和微软亚洲研究院李朝卓老师



中南大學
CENTRAL SOUTH UNIVERSITY



恳请老师批评指正!

答 辩 人：计科1901班 庄纹纹

指导老师：王 森 章