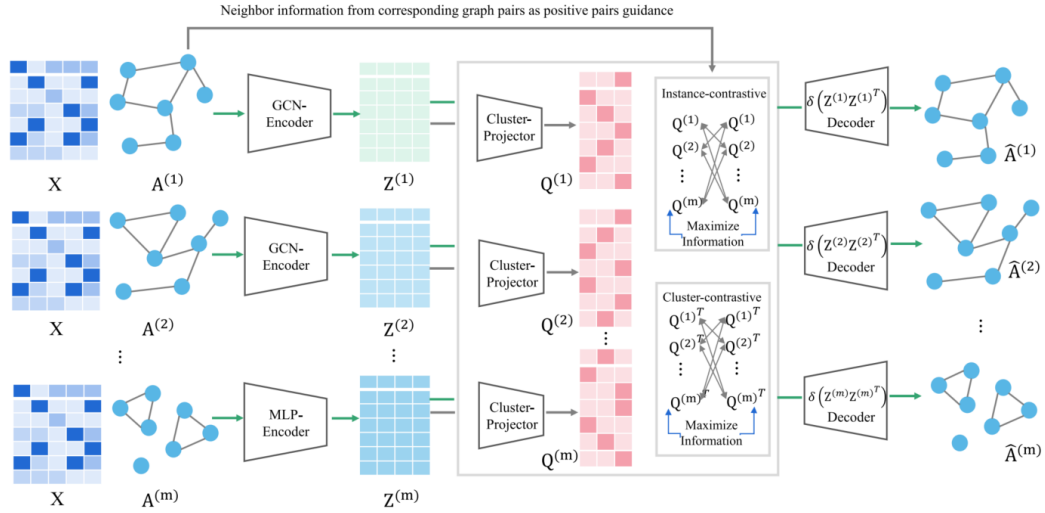


Contrastive Multiview Attribute Graph Clustering With Adaptive Encoders(CMAGC)

1. 背景:

本文选取的任务属于多视角属性图聚类，由多个视角或拓扑图结构和节点属性信息构成。每个视角可以表示不同的节点关系。虽然现在在某些任务上取得了一定进展，但现有方法没有充分考虑不同视角图的高阶邻居信息，并且大多使用相同的编码器（如GCN），没有针对不同视角的异质性进行自适应选择。并且很多方法无法有效处理稀疏图结构，导致聚类结果下将。因此本文引入了**自适应编码器**。

2.



引入的原因是如果图中不存在高阶邻居信息，即节点的所有高阶邻居与一阶邻居相同，那么使用GCN会导致信息过度平滑，节点间无法区分。所以文中提出的自适应编码器会在先判断是否存在高阶邻居信息，判断下面的公式：

$$HNI(G^{(v)}) = \sum_{i=1}^n |S_2(o_i|G^{(v)}) - S_1(o_i|G^{(v)})|$$

公式的含义是计算第 v 个图中的 o_i 节点的一阶邻居集合和二阶邻居集合（一阶邻居的邻居），若HNI大于0，则说明两个集合不相同，存在高阶邻居信息，采用GCN；反之，不存在，采用MLP。其中两个结构层数如下：

$$L^{(v)} = \begin{cases} \left\lceil \frac{1}{2} r(G_{\max}^{(v)}) \right\rceil, & \text{if } HNI(G^{(v)}) > 0 \\ 3, & \text{if } HNI(G^{(v)}) = 0. \end{cases}$$

GCN选取最大连通子图半径的一半并向上取整，MLP取固定层数。

当每个视角通过各自的编码器后得到各自对应的嵌入 Z ，为了更好的获得好的嵌入，引入了多视角的图重构，下面是重构损失函数：

$$\mathcal{L}_R = \sum_{v=1}^m \mathcal{L}_R^{(v)} = - \sum_{v=1}^m \sum_{i=1}^n \sum_{j=1}^n A_{ij}^{(v)} \log(\hat{A}_{ij}^{(v)}).$$

通过训练不断优化得到更好的嵌入表达。

3. 将嵌入映射到聚类空间，通过一个两层的MLP（图上指的是CLuster-Projector），生成每个视角的聚类分配矩阵 Q ，公式为：

$$Q^{(v)} = \text{softmax}(f^{(v)}(Z^{(v)}))$$

Q的维度是N*K维，表示每个节点属于不同聚类的概率

4. 为了保持多视角下的节点嵌入一致性，也是为了将各视角的嵌入更好的结合起来，使得同一节点在不同视角下的表示尽可能相似，捕捉到多视角的公共信息，保证多视角下的聚类分配结果能够一致，引入了多视角对比学习。对比学习从两个层次来介入。

特征层

特征层的目标是在嵌入空间内进行学习，使得相同节点在不同视角下的嵌入尽可能一致。具体方法是将每个节点的第一阶邻居视为**正样本对**，并通过正样本对的对比来优化嵌入表示。在特征层对比学习中，针对节点 o_i ，从不同视角 v 和 w 中选择该节点的一阶邻居，构造正样本对。具体来说，节点 o_i 在视角 v 和视角 w 中的邻居联合集 N_i 被视为正样本。损失函数如下：

$$\ell_f^{(vw)} = -\frac{1}{n} \sum_{i=1}^n \sum_{t \in N_i^{(vw)}} e^{\frac{\sin(q_i^{(v)}, q_t^{(w)})}{\tau_f}} + e^{\frac{\sin(q_i^{(v)}, q_{t' \neq i}^{(w)})}{\tau_f}} \times \log \frac{e^{\frac{\sin(q_i^{(v)}, q_t^{(w)})}{\tau_f}} + e^{\frac{\sin(q_i^{(v)}, q_{t' \neq i}^{(w)})}{\tau_f}}}{\sum_{t'=1}^n \sum_{u=v,w} e^{\frac{\sin(q_i^{(v)}, q_{t'}^{(u)})}{\tau_f}}}$$

q_i 是i节点在v视角的嵌入， \sin 余弦相似度(公式识别错了，应该是sim)

聚类层

聚类层对比学习是在**聚类空间**进行的，主要对比的是不同视角下的**聚类分配矩阵**。

$$\ell_c^{(vw)} = -\frac{1}{k} \sum_{j=1}^k \log \frac{e^{\frac{\sin(q_j^{(v)T}, q_j^{(w)T})}{\tau_c}}}{\sum_{h=1}^k \sum_{u=v,w} e^{\frac{\sin(q_j^{(v)T}, q_h^{(u)T})}{\tau_c}}}$$

聚类层对比学习中，节点 o_i 在视角 v 和视角 w 的聚类分配概率矩阵 $Q(v)$ 和 $Q(w)$ 被用作正样本对，目标是确保节点在不同视角下被分配到相同聚类的概率尽可能高。 q 是指聚类概率矩阵，模型中的 Q 。

总的损失函数：

$$\mathcal{L}_F = \frac{1}{2} \sum_{v=1}^m \sum_{w \neq v} \ell_f^{(vw)}.$$

$$\mathcal{L}_C = \frac{1}{2} \sum_{v=1}^m \sum_{w \neq v} \ell_c^{(vw)}.$$

$$\mathcal{L} = \mathcal{L}_R + \mathcal{L}_F + \mathcal{L}_C.$$

5. 对比学习完成之后，将各个视角得到的分配概率矩阵进行取均值融合，这样得到最终的概率矩阵就包含着各个视角的信息。文中并采用传统的聚类方法，应该就是直接拿分配概率当作最后的分配结果。

数据集：

Datasets	#Node	#Attribute dimension	#View	#Edges in each adjacent matrix	#Layers of encoder	#Clusters
ACM	3025	1830	2	co-subject (29,281) co-author (2,210,761)	$\left\lceil \frac{1}{2} r \left(G_{\max}^{(1)} \right) \right\rceil = 5$ 3	3
DBLP	4057	334	3	co-author (11,113) co-conference (5,000,495) co-term (6,776,335)	$\left\lceil \frac{1}{2} r \left(G_{\max}^{(1)} \right) \right\rceil = 6$ $\left\lceil \frac{1}{2} r \left(G_{\max}^{(2)} \right) \right\rceil = 1$ $\left\lceil \frac{1}{2} r \left(G_{\max}^{(3)} \right) \right\rceil = 1$	4
IMDB	4780	1232	2	co-actor (98,010) co-director (21,018)	$\left\lceil \frac{1}{2} r \left(G_{\max}^{(1)} \right) \right\rceil = 4$ 3	3

7. 结果：

((CMAGC − AVERAGE BASELINE)/AVERAGE BASELINE) × 100%												
Datasets	ACM				DBLP				IMDB			
Method	ACC	NMI	F1	ARI	ACC	NMI	F1	ARI	ACC	NMI	F1	ARI
LINE _{best} (2015)	0.6479	0.3941	0.6594	0.3433	0.8689	0.6676	0.8546	0.6988	0.4268	0.0031	0.2870	N/A
LINE _{avg} (2015)	0.6479	0.3941	0.6595	0.3432	0.8750	0.6681	0.8660	0.7056	0.4719	0.0063	0.2985	N/A
GAE _{best} (2016)	0.8216	0.4914	0.8225	0.5444	0.8859	0.6925	0.8743	0.7410	0.4298	0.0402	0.4062	0.0473
GAE _{avg} (2016)	0.6990	0.4771	0.7025	0.4378	0.5558	0.3072	0.5418	0.2577	0.4442	0.0413	0.4172	0.0491
ARGA (2018)	0.8433	0.5454	0.8451	0.6064	0.5816	0.2951	0.5938	0.2392	0.4119	0.0063	0.3685	N/A
SDCN (2020)	0.9045	0.6831	0.9042	0.7391	0.6805	0.3950	0.6771	0.3915	0.4697	0.0285	0.3183	0.0284
MNE (2018)	0.6370	0.2999	0.6479	0.2486	OM	OM	OM	OM	0.3958	0.0017	0.3316	0.0008
PMNE _n (2017)	0.6936	0.4648	0.6955	0.4302	0.7925	0.5914	0.7966	0.5265	0.4958	0.0359	0.3906	0.0366
PMNE _r (2017)	0.6492	0.4063	0.6618	0.3453	0.3835	0.0872	0.3688	0.0689	0.4697	0.0014	0.3183	0.0115
PMNE _c (2017)	0.6998	0.4775	0.7003	0.4431	OM	OM	OM	OM	0.4719	0.0285	0.3882	0.0284
RMSC (2014)	0.6315	0.3973	0.5746	0.3312	0.8994	0.7111	0.8248	0.7647	0.2702	0.0054	0.3775	0.0018
PwMC (2017)	0.4162	0.0332	0.3783	0.0395	0.3253	0.0190	0.2808	0.0159	0.2453	0.0023	0.3164	0.0017
SwMC (2017)	0.3831	0.0838	0.4709	0.0187	0.6538	0.3760	0.5602	0.3800	0.2671	0.0056	0.3714	0.0004
O2MAC (2020)	0.9042	0.6923	0.9053	0.7394	0.9074	0.7287	0.9013	0.7780	0.4502	0.0421	0.4159	0.0564
O2MA (2020)	0.8880	0.6515	0.8894	0.6987	0.9040	0.7257	0.8976	0.7705	0.4697	0.0524	0.4229	0.0753
MvAGC (2021)	0.8975	0.6735	0.8986	0.7212	0.9277	0.7727	0.9225	0.8276	0.5633	0.0371	0.3783	0.0940
SMC (2022)	0.8863	0.6397	0.8869	0.6929	0.9347	0.7862	0.9304	0.8421	0.5686	0.0356	0.4048	0.0863
MVGC _{Gabor} (2022)	0.8532	0.6327	0.8505	0.6308	0.8153	0.6249	0.8172	0.5795	0.4567	0.0163	0.3698	0.0368
MVGC _{Euler} (2022)	0.9617	0.8438	0.9608	0.8917	0.9233	0.7424	0.9225	0.8041	0.5331	0.0623	0.3953	0.0848
CMAGC*	0.9064	0.6785	0.9067	0.7417	0.9199	0.7639	0.9155	0.8075	0.5393	0.0609	0.4539	0.1068
CMAGC [#]	0.9002	0.6628	0.9002	0.7264	0.9046	0.7341	0.8997	0.7737	0.5333	0.0588	0.4505	0.0949
CMAGC	0.9369	0.7642	0.9373	0.8208	0.9367	0.7937	0.9331	0.8451	0.5768	0.0771	0.4193	0.1202
Improvement _{min}	−2.58%	−9.43%	−2.45%	−7.95%	0.21%	0.95%	0.29%	0.36%	6.95%	26.60%	−7.62%	12.55%
Improvement _{avg}	23.96%	51.09%	23.64%	60.88%	20.76%	41.08%	22.73%	46.34%	29.07%	183.40%	11.72%	157.40%

8. 思路：

本篇文章前面还是用了重构，将图结构重构，值得学习的地方是多视角下两个层面的对比学习是关键。