

- [Temporal Graph Learning](#)
 - [Temporal Heterogeneous Information Network Embedding\[IJCAI'21\]](#)
 - [EvolveGCN Evolving Graph Convolutional Networks for Dynamic Graphs\[AAAI'20\]](#)
 - [Temporal-Aware Graph Neural Network for Credit Risk Prediction\[SDM'21\]](#)
 - [T-GCN: A Temporal Graph Convolutional Network for Traffic Prediction\[TITS'17\]](#)
- [Graph Anomaly Detection](#)
 - [Deep Anomaly Detection on Attributed Networks\[SDM'19\]](#)
 - [GeniePath: Graph Neural Networks with Adaptive Receptive Paths\[AAAI'19\]](#)
 - [Decoupling Representation Learning and Classification for GNN-based Anomaly Detection\[SIGIR'21\]](#)
 - [AddGraph: Anomaly Detection in Dynamic Graph Using Attention-based Temporal GCN\[IJCAI'19\]](#)
 - [Characterizing and Forecasting User Engagement with In-app Action Graph: A Case Study of Snapchat](#)
 - [Structural Temporal Graph Neural Networks for Anomaly Detection in Dynamic Graphs\[CIKM'20\]](#)
 - [Anomaly Detection in Dynamic Graphs via Transformer\[TKDE'21\]](#)
- [Graph Self-Supervised Learning](#)
 - [Self-supervised Graph Learning for Recommendation\[SIGIR'21\]](#)
 - [Self-supervised Representation Learning on Dynamic Graphs\[CIKM'21\]](#)
 - [Multi-View Self-Supervised Heterogeneous Graph Embedding\[ECML-PKDD'21\]](#)
 - [Graph Debiased Contrastive Learning with Joint Representation Clustering\[IJCAI'21\]](#)
 - [Self-supervised Heterogeneous Graph Neural Network with Co-contrastive Learning\[KDD'21\]](#)
 - [GCC--Graph Contrastive Coding for Graph Neural Network Pre-Training\[KDD'20\]](#)
 - [GCCAD: Graph Contrastive Coding for Anomaly Detection](#)
 - [Deep Graph Infomax\[ICLR'19\]](#)
 - [InfoGraph: Unsupervised and Semi-supervised Graph-Level Representation Learning via Mutual Information Maximization\[ICLR'20\]](#)
 - [Spatio-Temporal Deep Graph Infomax\[ICLR'19 workshop\]](#)
 - [Self-Supervised Hypergraph Convolutional Networks for Session-based Recommendation\[AAAI'21\]](#)
 - [Self-Supervised Multi-Channel Hypergraph Convolutional Network for Social Recommendation\[WWW'21\]](#)
 - [Contrastive and Generative Graph Convolutional Networks for Graph-based Semi-Supervised Learning\[AAAI'21\]](#)
 - [Spatio-Temporal Graph Contrastive Learning](#)
 - [TCL: Transformer-based Dynamic Graph Modelling via Contrastive Learning](#)

Temporal Graph Learning

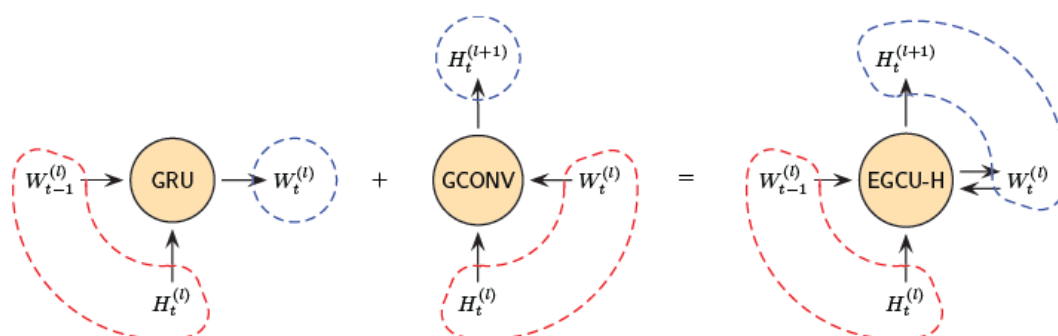
动态图学习相关论文，对每篇读过的论文给出基本介绍

Temporal Heterogeneous Information Network Embedding[IJCAI'21]

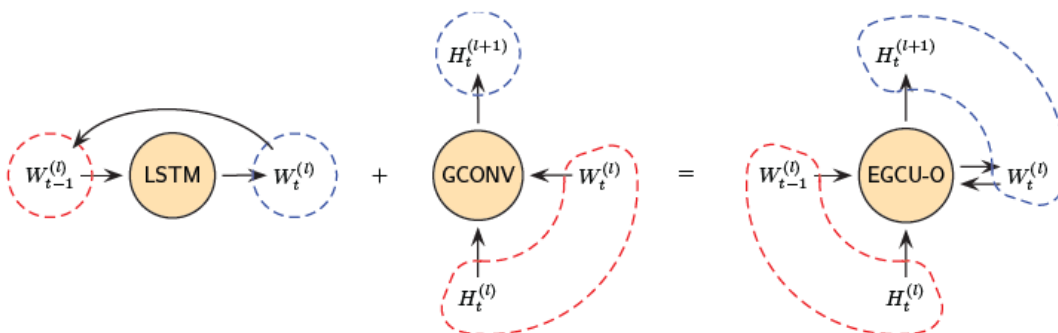
- 标签：动态图学习、异构图学习
- 摘要：对于异构信息（顶点和边），论文使用基于元路径的随机游走来处理。元路径能够描述起始顶点和终止顶点之间的复杂关系，它可以经过任意类型的顶点和边。对于动态信息，使用Hawkes过程来建模过去事件对当前事件的影响，以attention机制来区分每个事件影响的重要程度。
- 链接：<https://www.ijcai.org/proceedings/2021/203>
- 是否有开源代码：<https://github.com/S-rz/THINE>

EvolveGCN Evolving Graph Convolutional Networks for Dynamic Graphs[AAAI'20]

- 标签：动态图学习
- 摘要：将动态图看做一系列快照图，EvolveGCN将 $t+1$ 时刻与 t 时刻特征之间的关系通过GCN表示为 $H_t^{(l+1)} = \sigma(A_t H_t^{(l)} W_t^{(l)})$ ，关键在于如何更新 $W_t^{(l)}$ 。第一种方式将它看作网络的隐层表示，通过GRU进行更新；第二种方式将它看作网络的输出，通过LSTM进行更新。前者额外的将 $H_t^{(l)}$ 作为输入，而后者将上一时刻的输出直接作为输入：



(a) EvolveGCN-H, where the GCN parameters are hidden states of a recurrent architecture that takes node embeddings as input.



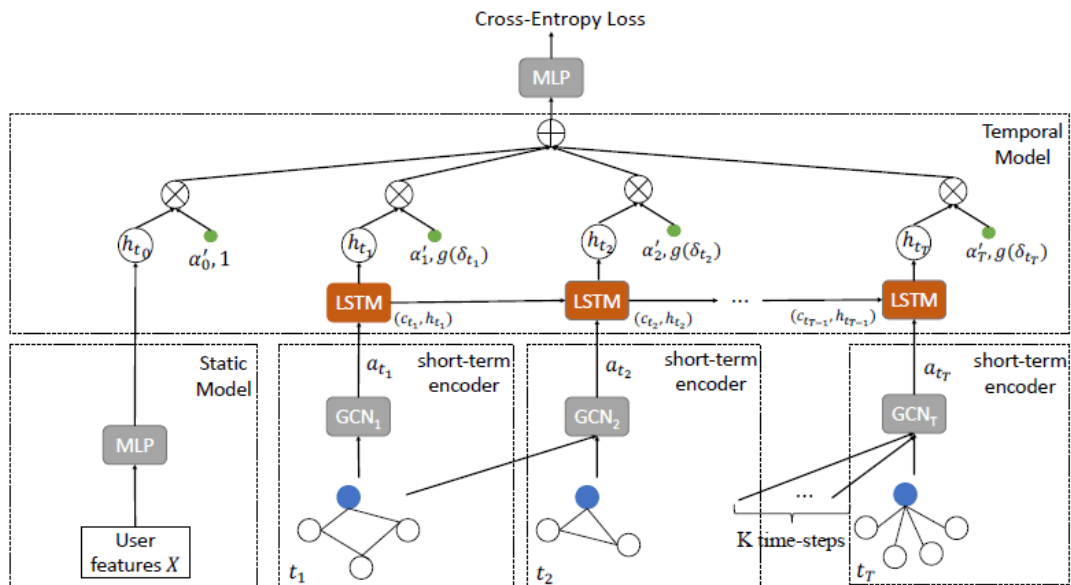
(b) EvolveGCN-O, where the GCN parameters are input/outputs of a recurrent architecture.

- 链接：<https://ojs.aaai.org/index.php/AAAI/article/view/5984/5840>
- 是否有开源代码：<https://github.com/IBM/EvolveGCN>

Temporal-Aware Graph Neural Network for Credit Risk Prediction[SDM'21]

- 标签：动态图学习
- 摘要：蚂蚁金服提出TemGNN模型用于信用卡欺诈检测。模型分为三部分，第一部分为MLP，用于将固定不变的特征如教育背景转换为静态的特征向量；第二部分为short-term图卷积，通过滑动窗口+attention聚合的形式对邻域信息进行聚合；第三部分为long-term编码，相较上一部分使用一个更长的滑动窗口，将形成的快照图序列通过LSTM，但TemGNN没有直接将LSTM的最终输出作

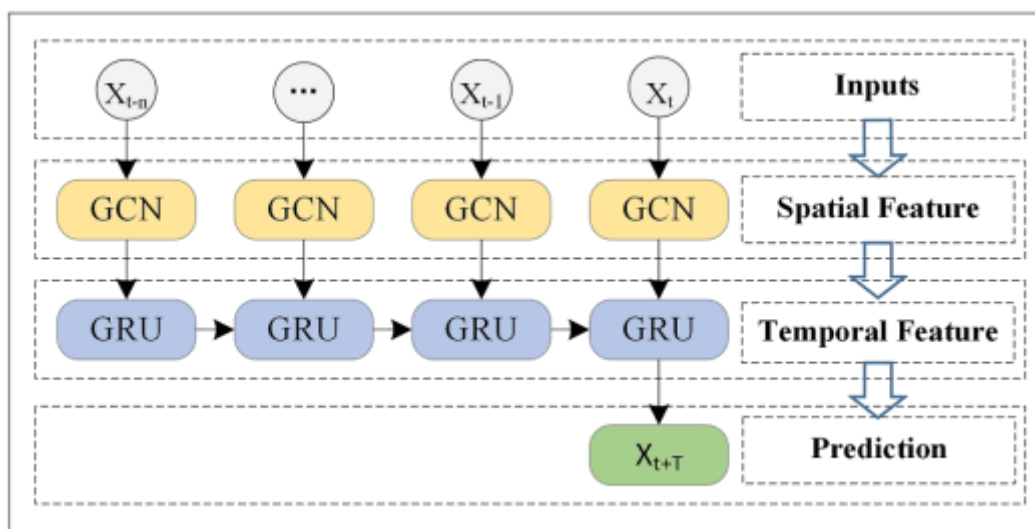
为结果，而是将每一个cell的输出通过一个attention的形式进行聚合，其中还加入了一个时间衰减因子，来减少久远事件对当前的影响。



- 链接: <https://epubs.siam.org/doi/abs/10.1137/1.9781611976700.79?mobileUi=0>
- 是否有开源代码: 无

T-GCN: A Temporal Graph Convolutional Network for Traffic Prediction[TITS'17]

- 标签:
- 摘要: T-GCN将GCN与GRU结合来捕获交通数据中的空间与时间信息，其中GCN用于学习空间上的拓扑结构而GRU用来学习时间上的动态变化。

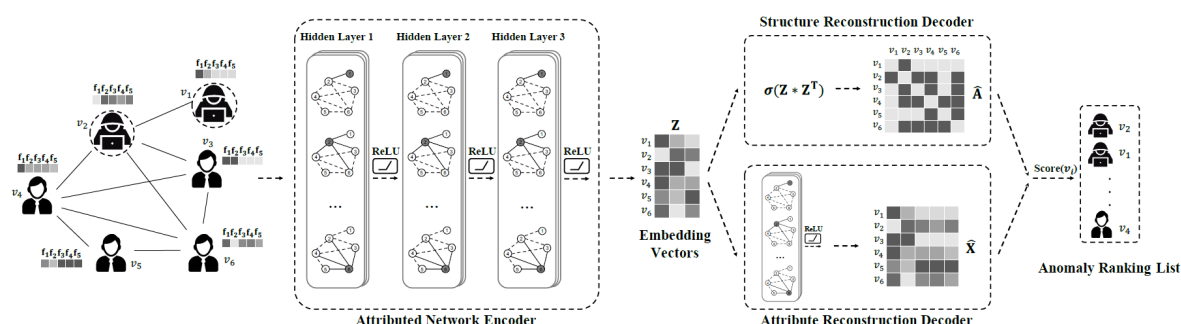


- 链接: <https://ieeexplore.ieee.org/document/8809901>
- 是否有开源代码: <https://github.com/lehaifeng/T-GCN/tree/master/T-GCN/T-GCN-PyTorch>

Graph Anomaly Detection

Deep Anomaly Detection on Attributed Networks[SDM'19]

- 标签:
- 摘要: 论文提出的DOMINANT方法十分直观, 是一个典型的Encoder-Decoder架构。首先通过GCN对输入图进行编码, 再将得到的顶点embedding通过向量点乘的形式去还原输入图的邻接矩阵与特征矩阵, 通过近似值与真实值之间的差值计算损失函数, 以及计算目标顶点的预测分数。

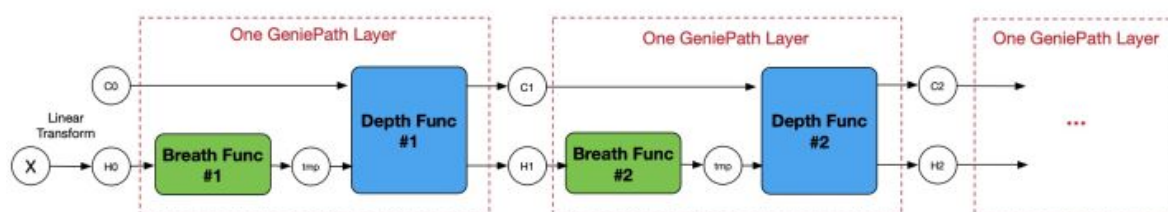


- 链接: <https://epubs.siam.org/doi/abs/10.1137/1.9781611975673.67?mobileUi=0>
- 是否有开源代码: https://github.com/kaize0409/GCN_AnomalyDetection_pytorch

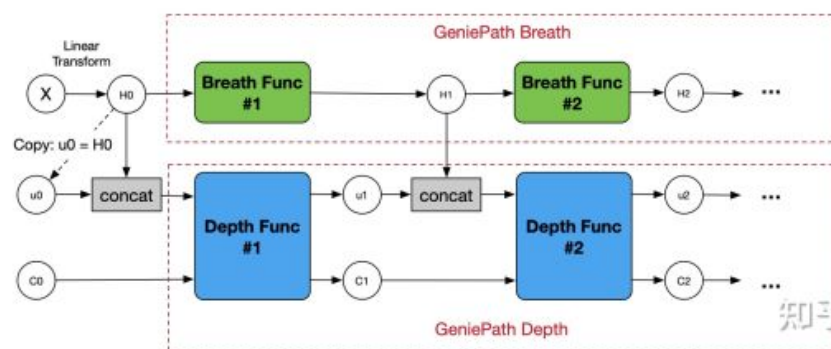
GeniePath: Graph Neural Networks with Adaptive Receptive Paths[AAAI'19]

- 标签:
- 摘要: 参考GeniePath是蚂蚁团队提出的一个在图上自适应地聚合邻域信息的方法, 每一层都由breadth和depth两部分组成, 前者通过GAT为一阶邻居赋予不同的重要性, 后者基于LSTM, 将t阶邻居看作t个时刻, 通过遗忘门和保留门来对高阶邻居的信息进行过滤, 以达到自适应的目的。

GeniePath-Standard



GeniePath-Lazy



知乎 @天雨粟

- 链接: <https://aaai.org/ojs/index.php/AAAI/article/view/4354/4232>
- 有无开源代码: <https://github.com/kayzliu/GeniePath>

Decoupling Representation Learning and Classification for GNN-based Anomaly Detection[SIGIR'21]

- 标签：聚类、图异常检测
- 摘要：论文提出了DCI方法，在DGI基础上加入了k-means聚类。DGI通过顶点-图的形式进行对比学习，来训练编码器，其中图的表示通过pooling得到。而DCI通过顶点-子图的形式进行对比学习，其中的子图由k-means得到，总的损失函数由各子图取平均得到。

Algorithm 1: Deep Cluster Infomax

Input : Graph $G = (V, A, X)$, Number of clusters K ,
Number of training epochs t , Number of
re-clustering epochs \bar{t} .

Output: Optimized GNN encoder g

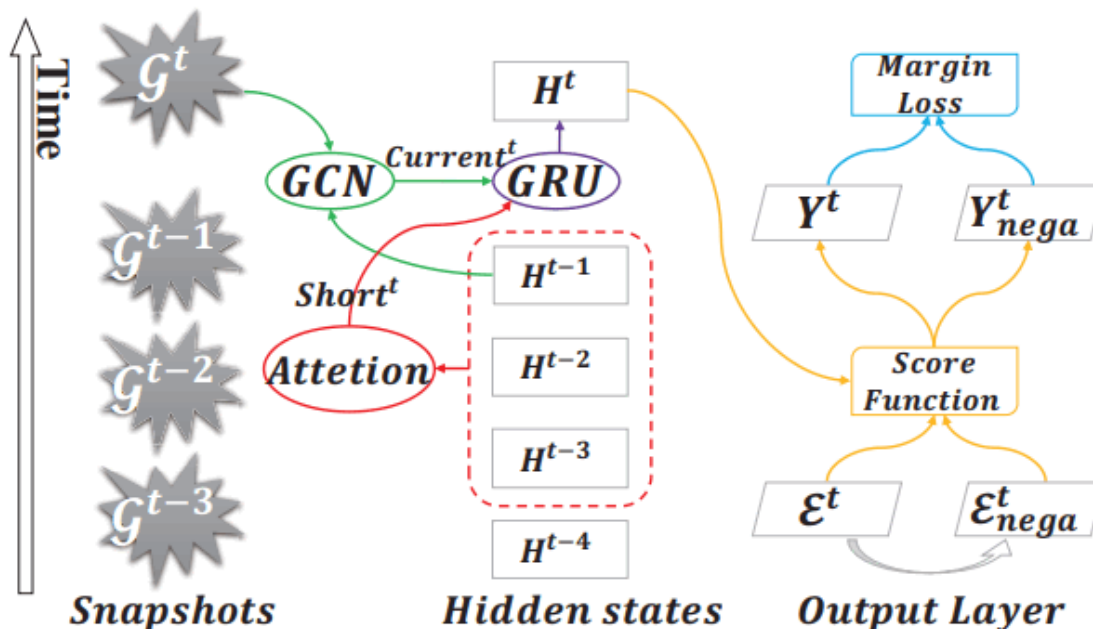
```
1 Initialize clusters  $[C_1, C_2, \dots, C_K] = \text{K-Means}(X)$ ;  
2 Initialize the parameters  $\theta$  and  $\omega$  for the encoder  $g$  and  
   the discriminator  $\mathcal{D}$  ;  
3 for  $epoch \leftarrow 1$  to  $t$  do  
4    $H = g(G, \theta)$ ;  
5    $\mathcal{L}_{DCI} = \frac{1}{K} \sum_{k=1}^K \mathcal{L}_{DCI}^k(H, C_k, \omega)$ ;  
6    $\theta, \omega \leftarrow \text{Adam}(\mathcal{L}_{DCI})$ ;  
7   if  $t \bmod \bar{t} == 0$  then  
8      $[C_1, C_2, \dots, C_K] = \text{K-Means}(g(G, \theta))$ 
```

Return: encoder g

- 链接: <https://dl.acm.org/doi/10.1145/3404835.3462944>
- 是否有开源代码: <https://github.com/wyl7/DCI-pytorch>

AddGraph: Anomaly Detection in Dynamic Graph Using Attention-based Temporal GCN[IJCAI'19]

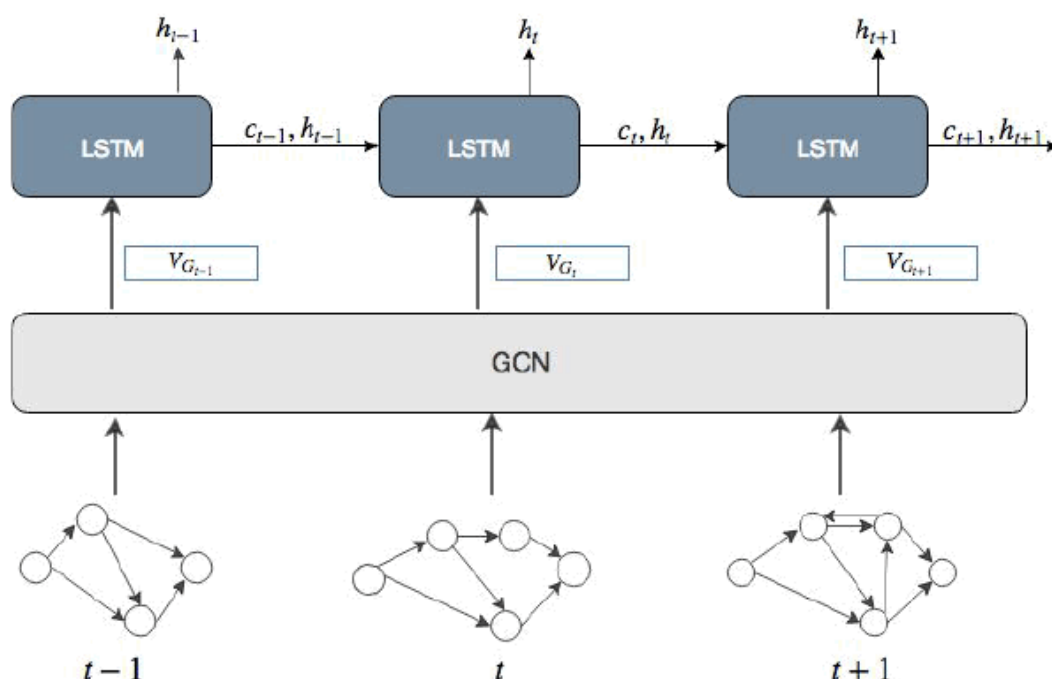
- 标签：动态图学习
- 摘要：论文的目的是检测图上的异常边。做法类似于T-GCN，对于一个包含若干快照的动态图，为了得到顶点在 t 时刻的特征表示，AddGraph首先将 $t-1$ 时刻的特征表示通过一个多层的GCN，得到长期的表示，再将 t 时刻的顶点通过一个基于attention的上下文模型来得到短期的表示，将长短期两部分输入一个GRU来得到 t 时刻的特征表示 H^t ，随后被用来计算两个顶点间出现异常边的概率。因为有标签数据的获取难度，论文还加入了SVM中的Hinge损失函数来进行负采样。



- 链接: <https://www.ijcai.org/proceedings/2019/614>
- 是否有开源代码: <https://github.com/Ljiajie/Addgraph>

Characterizing and Forecasting User Engagement with In-app Action Graph: A Case Study of Snapchat

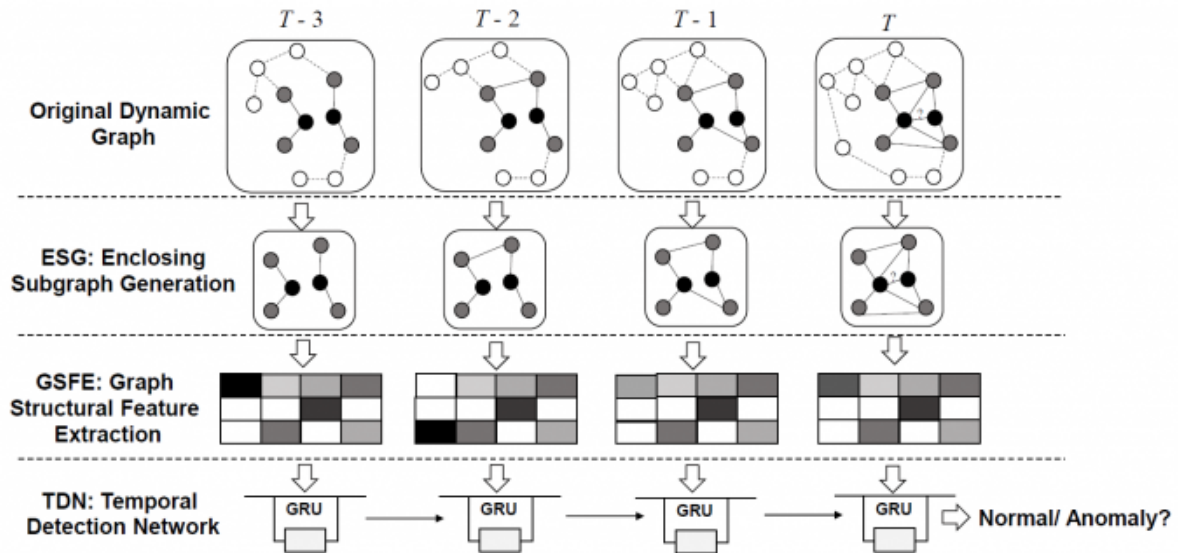
- 标签: 动态图学习
- 摘要: 在建模动态信息时, 论文将GCN与LSTM相结合, 先通过GCN为每个时刻的快照图生成 embedding, 所有时刻的快照图embedding就形成了一条序列, 再将这条序列输入LSTM中, 取最后一层的隐层输出为最终的特征表示。



- 链接: <https://dl.acm.org/doi/10.1145/3292500.3330750>
- 是否有开源代码: <https://github.com/INK-USC/temporal-gcn-lstm>

Structural Temporal Graph Neural Networks for Anomaly Detection in Dynamic Graphs[CIKM'20]

- 标签：动态图学习
- 摘要：论文提出的StrGNN用于异常边检测，分为三个部分：第一部分以边为中心提取h-hop子图，并根据结构信息对子图中的顶点进行标记；第二部分通过GNN为上一步中的子图编码，并通过Sort pooling的方式只取子图中的重要性前k个顶点形成子图的特征表示；第三部分通过GRU将各时刻特征表示形成的序列合成为t时刻目标边的特征表示。



- 链接：<https://dl.acm.org/doi/pdf/10.1145/3459637.3481955>
- 是否有开源代码：无

Anomaly Detection in Dynamic Graphs via Transformer[TKDE'21]

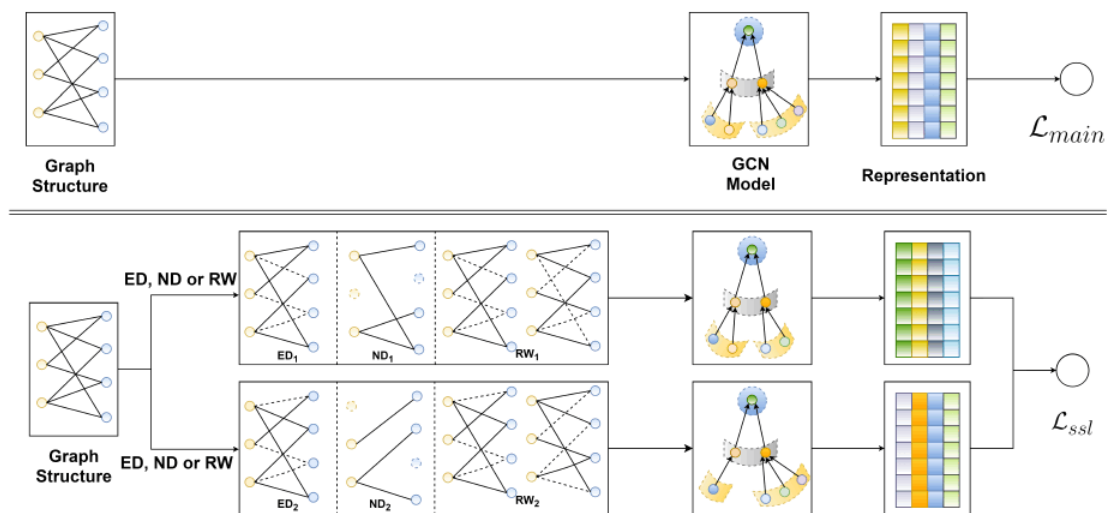
- 标签：
- 摘要：论文提出了TADDY方法，将Transformer应用于动态图的异常边检测问题。TADDY分为四个部分，第一部分为以边为中心提取子图，不同于StrGNN采用h-hop的形式，TADDY通过graph diffusion如PPR的形式为图中顶点计算connectivity，取目标边的头尾顶点中connectivity值top-k的顶点来构造子图。第二部分为三个顶点编码方案，为了解决动态图中特征缺失的问题。
- 链接：<https://shiruiipan.github.io/publication/tkde-liu-21/>
- 是否有开源代码：https://github.com/yuetan031/TADDY_pytorch

Graph Self-Supervised Learning

图上自监督学习相关论文，对每篇读过的论文给出基本介绍

Self-supervised Graph Learning for Recommendation[SIGIR'21]

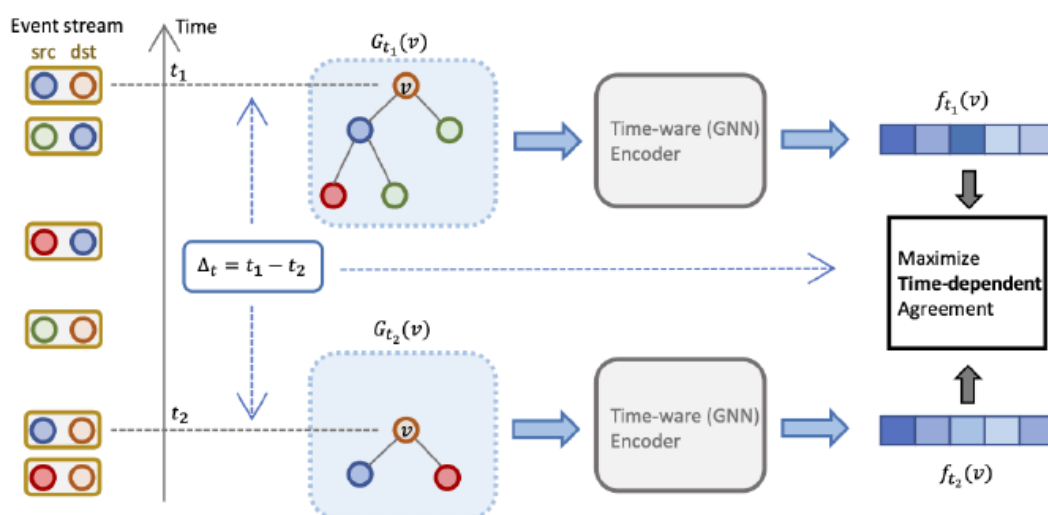
- 标签：图自监督学习
- 摘要：在用户-物品交互图上应用对比学习，提出了图上三种数据增强的方式：顶点dropout、边dropout以及随机游走，来生成同一顶点的不同视角，最小化同一顶点不同视角之间的差异，同时最大化不同顶点的视角之间的差异。论文花了较大篇幅对加入对比学习导致的性能提升给出了理论分析。



- 链接: <https://dl.acm.org/doi/abs/10.1145/3404835.3462862>
- 是否有开源代码: <https://github.com/wujcan/SGL>

Self-supervised Representation Learning on Dynamic Graphs[CIKM'21]

- 标签: 图自监督学习、动态图学习
- 摘要: 论文提出了一个去偏的动态图对比学习框架DDGCL。对于动态信息的处理, DDGCL从几个方面展开: 首先在将子图转化成embedding时, 在GNN中加入了时间 t 的embedding。其次, DDGCL从过去一个给定的时间间隔内随机取一个时刻对应的子图作为正样本, 认为这段时间内顶点的标签没有发生变化。最后, 在进行对比学习时, 用了一个时间相关的相似度函数, 对时间间隔进行了惩罚, 避免两个子图结构相似的顶点在较长的时间间隔下也被判断为相似。

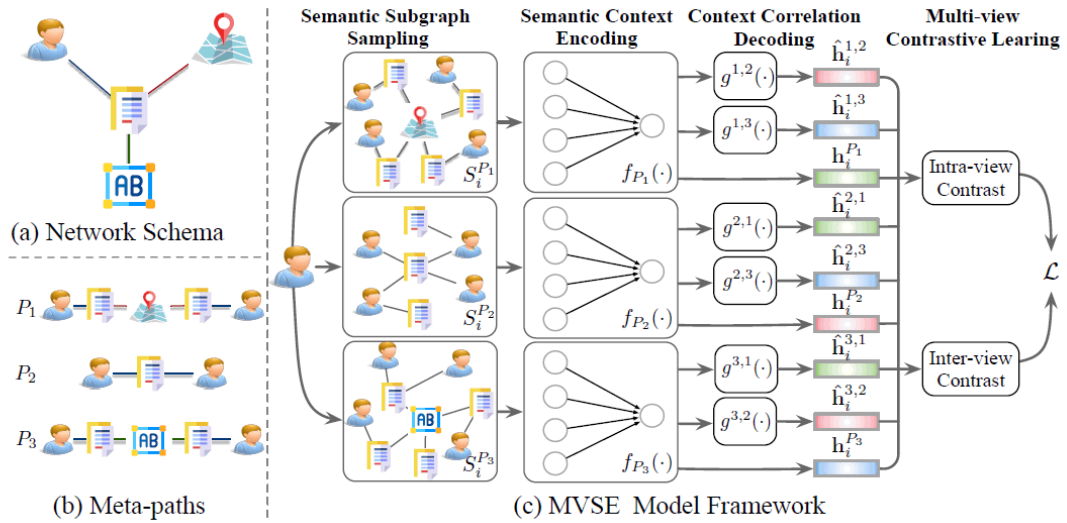


- 链接: <https://dl.acm.org/doi/abs/10.1145/3459637.3482389>
- 是否有开源代码: 无

Multi-View Self-Supervised Heterogeneous Graph Embedding[ECML-PKDD'21]

- 标签: 元路径

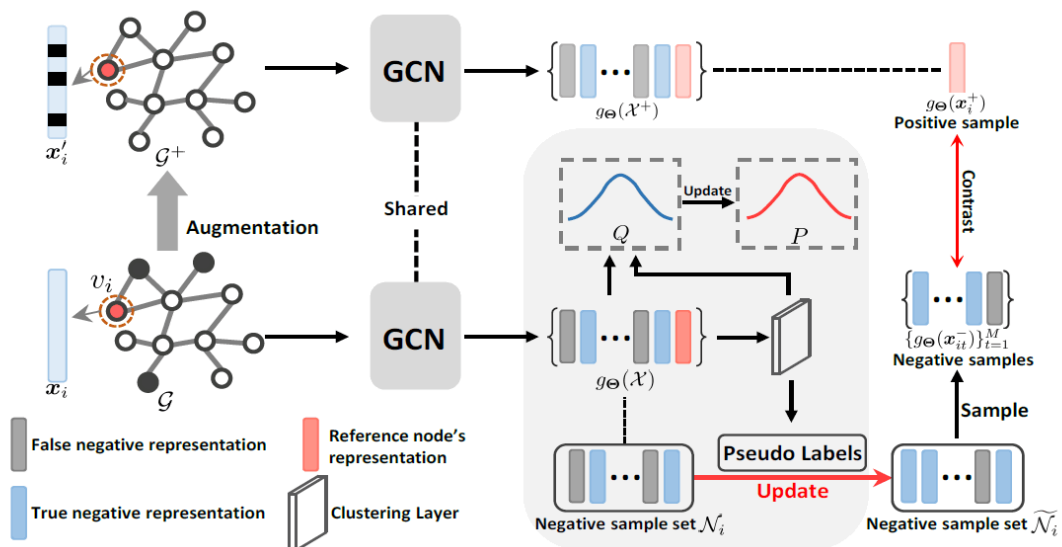
- 摘要：给定异构图中一个顶点和关联的若干条元路径，论文首先对每条元路径通过固定长度的随机游走来获得元路径相关的子图，并通过GIN来得到每个子图的embedding。因为作者认为不同的元路径代表不同视角，所以顶点所形成的多个子图的embedding就构成了多视角的embedding，作为下一步MoCo对比学习框架输入的key和query。这里的对比学习包括intra-view和inter-view两种方式。



- 链接：https://2021.ecmlpkdd.org/wp-content/uploads/2021/07/sub_408.pdf
- 是否有开源代码：<https://github.com/Andy-Border/MVSE>

Graph Debiased Contrastive Learning with Joint Representation Clustering[IJCAI'21]

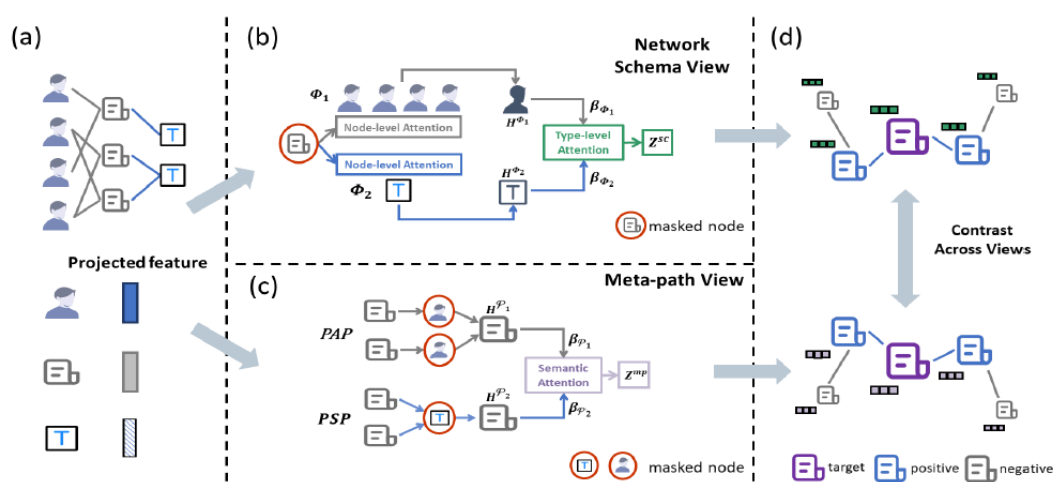
- 标签：聚类
- 摘要：现有的对比学习方法大多直接在图上随机采样作为负样本，这种做法是有偏的因为可能采到假阴性样本从而导致性能下降。论文提出的去偏方法是通过聚类来将不同簇的顶点作为负样本，减少图上随机采样的偏差。具体地，正样本是通过将输入顶点的特征随机mask，而负样本是通过将输入顶点经过GCN得到的embedding聚类后，选取不同簇的顶点。因此，聚类结果的好坏会很大影响模型的性能，论文使用KL散度与t分布来对每次迭代的聚类结果进行评估及更新。



- 链接：<https://www.ijcai.org/proceedings/2021/473>
- 是否有开源代码：无

Self-supervised Heterogeneous Graph Neural Network with Co-contrastive Learning[KDD'21]

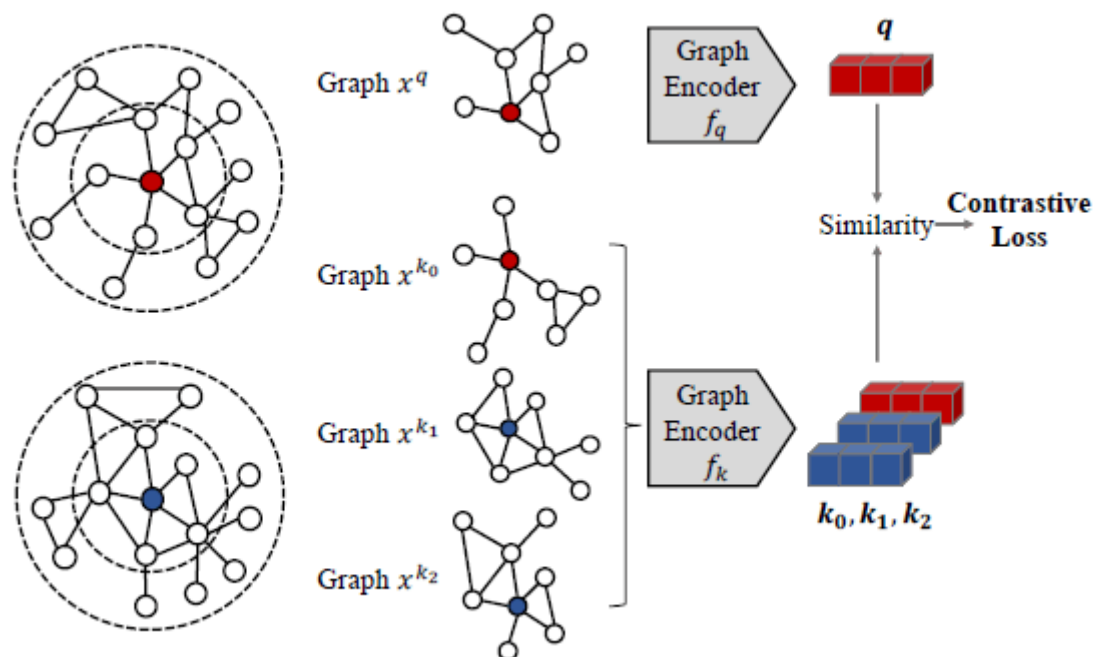
- 标签：元路径、attention
- 摘要：HeCo通过一个异构图的元路径和网络模式两个角度来学习顶点的embedding。首先将异构顶点的特征先投影到同一个特征空间中，得到顶点的初步embedding，接下来通过node-level和type-level的attention来对邻域内的异构顶点信息进行聚合，得到网络模式角度的embedding；先通过GCN将元路径上顶点的embedding聚合成元路径的embedding，再通过semantic-level的attention来得到元路径角度的embedding。得到两个角度的embedding后，HeCo将它们通过一个MLP来进行接下来的对比学习。HeCo的正负样本选取方式与其他图对比学习不同，它认为有多条元路径相连的两个顶点构成一对正样本，而元路径数量少于设定阈值的两个顶点视为一对负样本。



- 链接：<https://arxiv.org/abs/2105.09111>
- 是否有开源代码：<https://github.com/liun-online/HeCo>

GCC--Graph Contrastive Coding for Graph Neural Network Pre-Training[KDD'20]

- 标签：随机游走
- 摘要：GCC作为图对比学习的一个方法，提出了自己的构造正负样本的方式。GCC以子图为单位来构造样本，这里的子图定义为r中心网络，即与中心顶点最短距离不大于r的顶点所形成的集合。在构造正负样本时，给定一个中心顶点，通过带重启的随机游走->子图抽取->匿名化三个步骤得到它的正样本，将不同顶点得到的视为负样本。得到的子图样本经过GIN得到各自的embedding，通过InfoNCE进行对比学习。



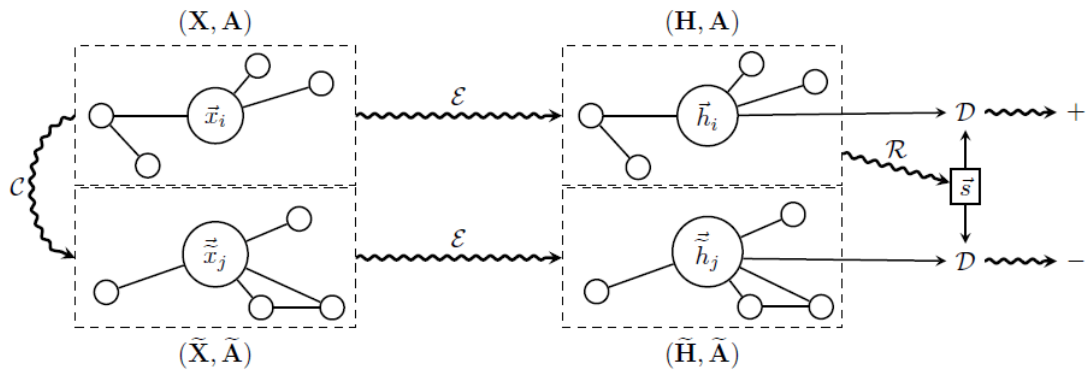
- 链接: <https://www.kdd.org/kdd2020/accepted-papers/view/gcc-graph-contrastive-coding-for-graph-neural-network-pre-training>
- 是否有开源代码: <https://github.com/THUDM/GCC>
- 数据量级: LiveJournal[n=484,3953]

GCCAD: Graph Contrastive Coding for Anomaly Detection

- 标签:
- 摘要:
- 链接: <https://arxiv.org/pdf/2108.07516.pdf>
- 是否有开源代码: <https://github.com/allanchen95/GCCAD>

Deep Graph Infomax[ICLR'19]

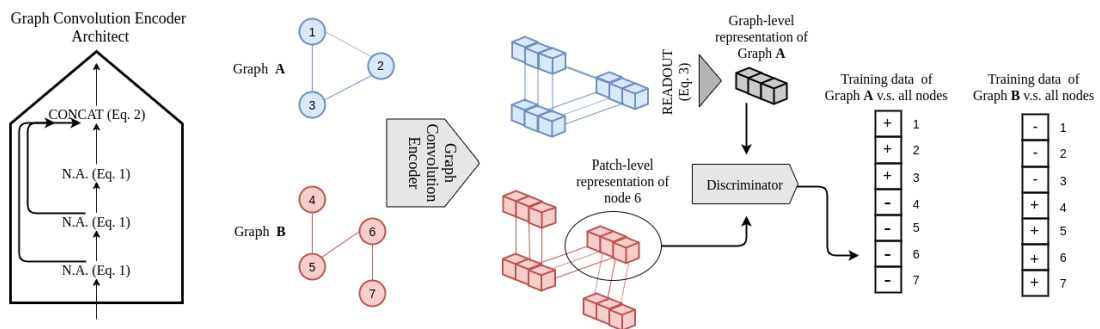
- 标签: 互信息
- 摘要: DGI将Infomax准则用在了图上, 所谓的Infomax准则是指最大化系统输入与输出之间的互信息。DGI通过最大化顶点与图特征表示之间的互信息来进行对比学习。输入一张图, DGI通过一个编码器例如GCN得到图中顶点的特征表示 H 后, 利用一个readout函数例如average pooling聚合顶点的特征表示得到图的特征表示 S 。顶点和图的特征表示彼此构成正样本对, 而对原图进行扰乱后经过相同编码器得到的顶点的特征表示 H' 表示则与原图的特征表示 S' 构成负样本对。按照Infomax准则, DGI中的系统指的是readout函数, 输入为顶点特征 H 输出为图的特征表示 S 。用于区分正负样本的判别器选取双线性评分函数 $D(h_i, s) = \sigma(h_i^T W s)$, 输出的是 (h_i, s) 为正样本对的概率, 用于更新特征表示的损失函数是Jensen-Shannon散度。关于DGI更详细的解析可见:
 - [互信息及其在图表示学习中的应用](#)
 - [深度学习中的互信息: 无监督提取特征](#)



- 链接: <https://openreview.net/forum?id=rklz9iAckQ>
- 有无开源代码: <https://github.com/PetarV-/DGI>
- 数据量级: Reddit[n=23,1443]

InfoGraph: Unsupervised and Semi-supervised Graph-Level Representation Learning via Mutual Information Maximization[ICLR'20]

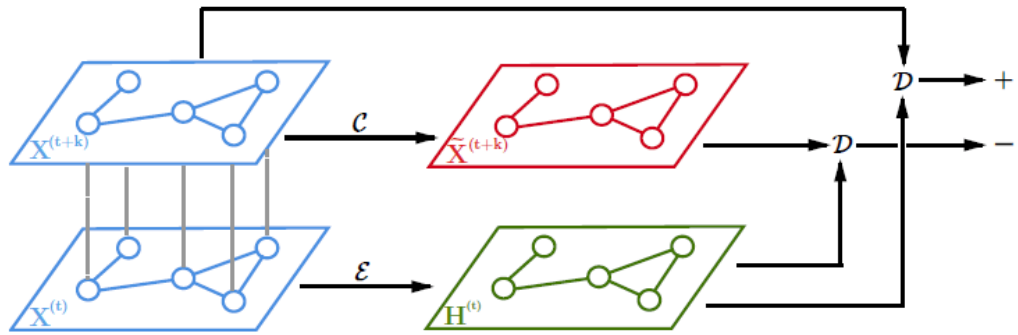
- 标签: 互信息
- 摘要: InfoGraph的输入是一系列图, 无监督任务下为每个图学习一个特征表示, 半监督任务下利用给定的有限带标签图为未知图预测。InfoGraph的无监督学习部分与DGI类似, 编码器上它使用GIN而不是DGI中的GCN, 将 K 层后GIN得到的各顶点的特征表示通过readout函数聚合后得到图的特征表示 S , 而对GIN每一层得到的顶点特征表示进行拼接, 得到patch特征表示作为正样本 H 。 S 表示全局信息而 H 表示局部信息。在负样本的选择上, InfoGraph将当前图的patch与不同图的特征表示进行组合作为负样本。在半监督任务下, 为了更好地进行知识迁移, 同时最大化监督部分的编码器与无监督部分的编码器之间的互信息。



- 链接: <https://arxiv.org/abs/1908.01000>
- 有无开源代码: <https://github.com/fanyun-sun/InfoGraph>
- 数据量级: RDT-M5K[n=4999*508.52]

Spatio-Temporal Deep Graph Infomax[ICLR'19 workshop]

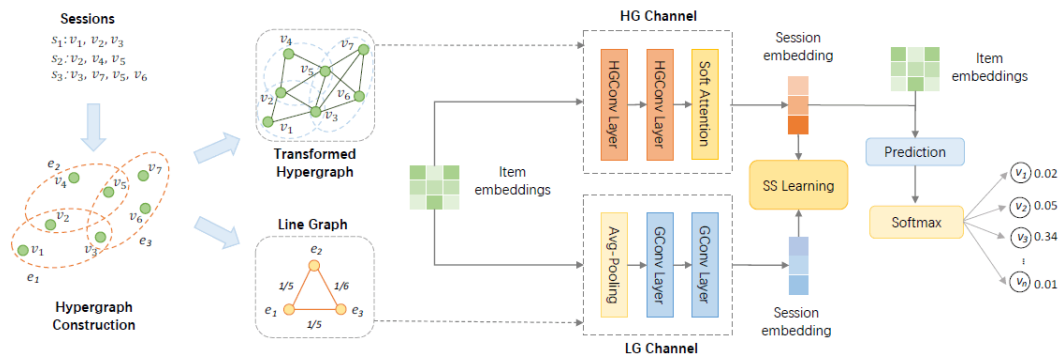
- 标签: 互信息、动态图学习
- 摘要: STDGI在DGI的基础上加入动态的时间信息, 体现在正负样本的构造上: 对于 t 时刻的图上所有顶点, 通过一个编码器如GCN得到它们的embedding $H^{(t)}$, 与 $t+k$ 时刻的顶点特征 $X^{(t+k)}$ 构成正样本; 与 $t+k$ 时刻随机mask的顶点特征构成负样本



- 链接: <https://rlgm.github.io/papers/48.pdf>
- 有无开源代码: 无
- 数据量级: METR-LA[n=207]

Self-Supervised Hypergraph Convolutional Networks for Session-based Recommendation[AAAI'21]

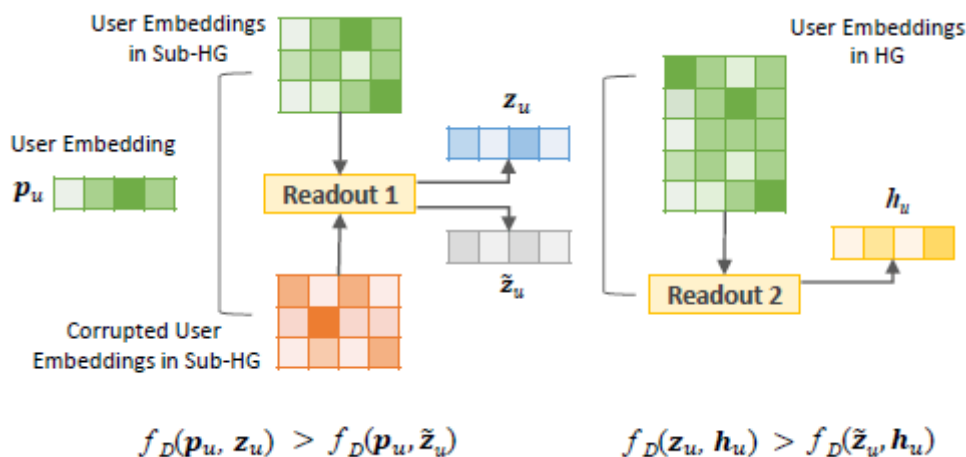
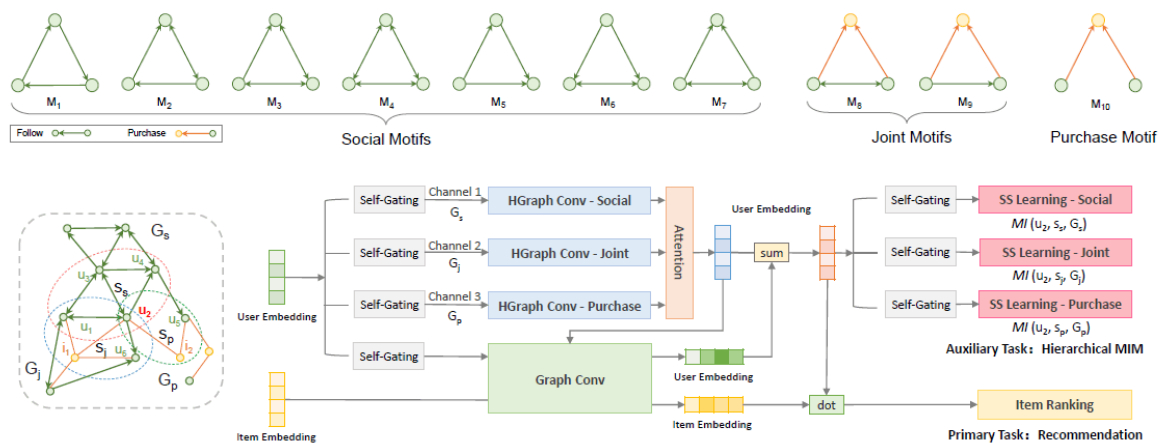
- 标签: 超图学习
- 摘要: 论文在序列推荐任务上提出了DHCN方法, 结合了超图学习和图对比学习的理论。所谓的超图是指将子图看作顶点, 在子图之间添加边, 构成一张超图。DHCN将一条用户点击序列看作一个子图, 序列内任意两个物品之间构成一条边。这样构成了下图中的"Transformed Hypergraph", 以物品为顶点; 而序列之间共享的物品作为连接两个子图的边。构成了下图中的"Line Graph", 以子图为顶点。两个图就构成了同一条序列的两个视角, 通过图卷积得到各自的embedding后, 同一序列的两个视角构成正样本, 不同序列视为负样本。判别器选择将两个输入的embedding点乘的形式, 损失函数为InfoNCE。



- 链接: <https://ojs.aaai.org/index.php/AAAI/article/view/16578/16385>
- 有无开源代码: <https://github.com/xiaxin1998/DHCN>
- 数据量级: Nowplaying[n=91,5128]

Self-Supervised Multi-Channel Hypergraph Convolutional Network for Social Recommendation[WWW'21]

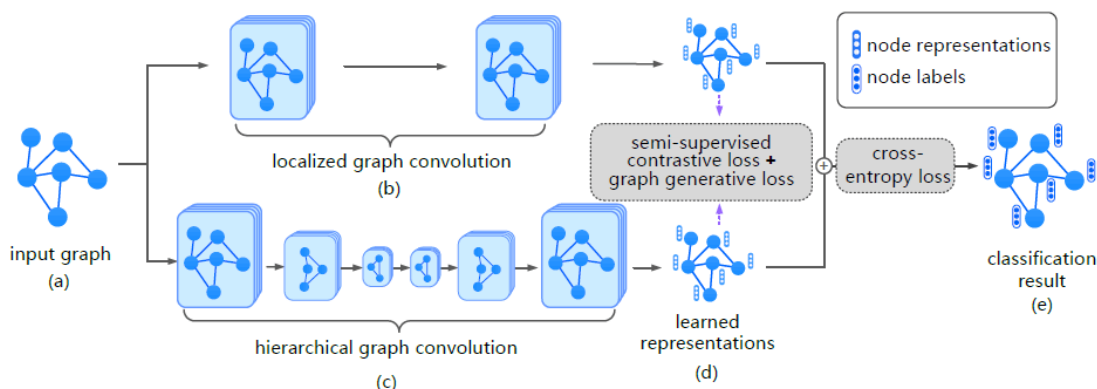
- 标签: 超图学习、Motif
- 摘要: 论文在社会推荐任务上提出了MHCN方法, 根据社会推荐场景中的好友-好友、好友-商品、用户-商品三种典型的关系模式, 通过三类十种三角Motif来构造三个视角的超图进行对比学习。不同于DGI直接进行顶点-图的互信息最大化, MHCN加入了以顶点为中心的子图来进行顶点-子图-图的层次化互信息最大化。构造正负样本时, 将子图中顶点的embedding顺序随机打乱来得到扰乱子图, 作为顶点-子图和子图-图两个阶段对比学习的负样本, 判别器选取的是将输入的两个embedding进行点乘, 损失函数不同于常见的交叉熵选取了排序损失。



- 链接: <https://dl.acm.org/doi/fullHtml/10.1145/3442381.3449844>
- 有无开源代码: <https://github.com/Coder-Yu/RecQ>
- 数据量级: Yelp[n=1,9539]

Contrastive and Generative Graph Convolutional Networks for Graph-based Semi-Supervised Learning[AAAI'21]

- 标签:
- 摘要: 论文解决的是如何利用有限带标签的样本来为大量无标签样本预测。分为无监督和有监督任务, 从局部和全局两个视角进行对比学习, GCN因任务为是对邻域信息聚合, 所以作为局部视角下的编码器, 而Hierarchical GCN用来产生全局视角。在无监督任务下, 因为没有数据的标签信息, 论文将同一顶点的两个视角看作正样本, 不同顶点视为负样本。在有监督任务时, 同一类的为正样本而不同类的为负样本。

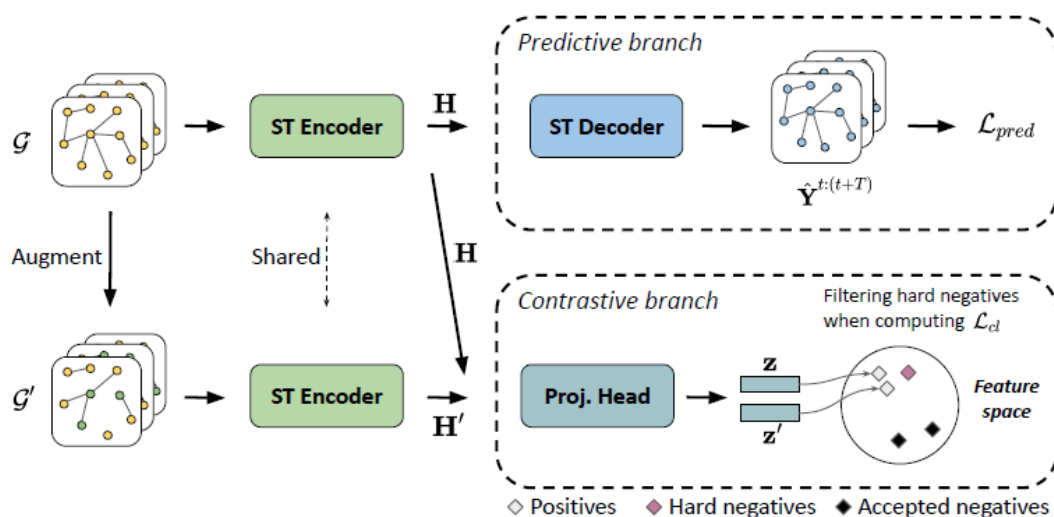


- 链接: <https://ojs.aaai.org/index.php/AAAI/article/view/17206/17013>
- 有无开源代码: <https://github.com/LEAP-WS/CG3>

- 数据量级: PubMed[n=1,9717]

Spatio-Temporal Graph Contrastive Learning

- 标签: 动态图学习
- 摘要: STGCL的输入是传感器的采集数据, 在构建正样本时, 除了传统的特征mask和边dropout, 还对相邻时刻的特征进行线性插值; 在构建负样本时, 为了避免采样到临近时刻的“hard negative”样本, 论文只考虑时间间隔大于给定阈值的负样本。



- 链接: <https://arxiv.org/abs/2102.12380>
- 有无开源代码: <https://github.com/Mobzhang/PT-DGNN>

TCL: Transformer-based Dynamic Graph Modelling via Contrastive Learning

- 标签: 动态图学习
- 摘要: TCL的目标是利用历史信息预测 t 时刻用户是否会与物品产生交互, 使用动态依赖-交互图作为图的呈现, 各自得到用户和物品的 k 最短距离子图后, 通过BFS转化为序列后输入定制的Transformer中, 得到各自的特征表示。在构建正负样本时, 用户已交互过的物品作为正样本, 其余未交互过的物品作为负样本
- 链接: <https://arxiv.org/abs/2105.07944>
- 有无开源代码: 无