

AdaMotif: 基于自适应 Motif 设计的图形简化

摘要

随着图形尺寸的增加,在有限的屏幕空间内,图形结构的清晰可视化变得困难甚至不可能。因此,设计有效的大规模图形可视化表示是非常关键的。本文提出了一种新的基于 AdaMotif 的大规模图结构分析方法,该方法通过自适应 Motif 设计,能够有效地提取大规模图的基本结构模式,并揭示大规模图的整体结构。具体地说,AdaMotif 方法包括将一个给定的大图划分为多个子图,然后对相似的子图进行聚类,并在每个聚类中提取相似的结构信息。随后,生成表示每个聚类的自适应 Motif,并利用该自适应 Motif 来替换对应的子图,从而得到简化的可视化。方法主要是为了保留子图中尽可能多的信息,同时有效地简化图。值得注意的是,这个方法能成功地将关键的社区信息可视化在一个大的图中。最后,以实际的图形进行案例研究及使用者研究,以验证提出方法的有效性。实验结果表明,该方法在简化图的同时保留了重要的结构和社区信息。

关键词: 大规模图; AdaMotif; 相似性布局; 差异性布局

1 引言

图是关系数据的一种常见形式,由节点和边组成。图的示例包括社交网络、网络拓扑和分子结构。随着图表规模的增加,分析大规模图在数据可视化和数据挖掘领域都提出了重大挑战。

在图形可视化中,节点-链接图方法由于其直观有效的表示而被广泛使用。然而,随着图的规模增加,相应的节点链接图变得更大规模,并且图中的结构信息也变得更复杂。有效的可视化方法可以帮助用户分析大型图中的结构信息。一个可行的方法是简化大规模的节点链接图。目前,简化大型图的研究主要集中在两个方向:基于字形的简化(例如 Motif 简化 [11] 和图采样 [26, 43, 46])。基于字形的简化抽象了图的结构,而图采样减少了节点的数量,以实现简化。Motif 简化可以通过使用需要较少屏幕空间的 Motif 来表示特殊的图形结构。因此,它可以用于分析大型图形,并具有广泛的实际应用 [11],例如,社交媒体网络¹。然而,这些 Motif 是预定义的并且只能应用于三种特定结构(即, *fan*, *connector* 和 *clique*)而不是一般结构。图形采样用于简化分析和处理大型网络的数据的可管理的子集。该技术可以减小图的大小,但不能突出图中的社区结构。

图中的社区信息具有重要的实际应用。例如,在社交网络中,可以利用社区信息来有效监控舆论发展,并向用户推荐合适的产品 [5]。通过比较社区之间的异同,可以制定和实施更精确的政策,提高政策的有效性。因此,在社会网络分析 [32, 38]、信息检索 [24]、推荐系统 [17, 44]、

¹<https://nodexl.com/>

运输和物流 [18] 以及生物网络分析 [10] 等领域中，提取和分析社区结构以及它们之间的相似性和差异性是非常重要的。

在图挖掘领域中，图中的一般社区结构可以通过子图划分 [9, 42] 和图嵌入 [33, 37] 来检测，并且可以通过图对齐来比较 [6, 22, 23, 34]。这些方法有效地帮助用户从图中挖掘社区信息，但缺乏直观的表达，阻碍了用户进一步的探索和理解。

将图形可视化与相关的图形挖掘技术相结合，提出了一种新的方法，通过自适应 *Motif* 设计的大型图形简化。*AdaMotif* 方法首先将图划分为子图（即，社区），从而产生许多子图块。然后，它将相似的子图聚类在一起，并计算每个聚类的质心子图，它表示该类子图的结构信息。最后，它对齐代表子图，以获得它们的相似结构，从而生成一个简化的主题，代表每个子图的整体结构。通过用简化的图案替换原始子图，显著降低了原始图的可视化的视觉复杂度，大大提高了视觉清晰度和可读性。与其他图简化方法类似，*AdaMotif* 方法也会导致信息丢失，这是不可避免的。虽然它丢失了不同社区中节点之间的原始边缘信息，但它非常适合分析社区的整体拓扑结构以及它们之间的异同。此外，它并不局限于分析特定的社区结构，可以自动生成适合于一般社区结构的主题，帮助用户直观地分析图的特征。

2 相关工作

2.1 图形可视化简化

现有的图形可视化简化研究主要分为基于字形的简化和基于图形采样的简化。基于字形的简化通过抽象图的结构来实现简化，而图采样通过直接减少节点的数量来实现简化。

基于字形的简化使用专门的图符号来表示相应的数据是一种有效的可视化方法，可以在不同的数据集上应用 [4, 14, 41]。*Motif* 被定义为在给定网络中出现的频率比预期的更高的子图模式 [1]。*Motif* 简化 [11] 利用图形表示来简化图形的可视化，通过设计特殊的 *Motif* 来增强图形的可读性，特别是那些具有三种特定结构的图形（即，*fan*, *connector* 和 *clique*）而不是一般结构。除了 *Motif* 简化之外，其他简化方法包括基于节点或边权重来合并节点 [35, 36, 45]。这些方法在有效地缩小图的规模、增强关键信息的同时，也会导致原始图中的重要结构信息丢失。

图形采样。图采样方法常用于减小图的规模，从而简化图的可视化 [8]。设计有效的抽样方法可以在减少视觉混乱的同时保留图的核心重要结构。MCGS [43] 方法对少量特殊节点进行采样以保持图的关键结构。其他图形采样方法包括上下文感知 [46] 和分层采样 [26] 等。然而，随着采样率的降低，来自图形的更多信息会丢失。

虽然 *AdaMotif* 方法也经历了信息丢失，但强调社区结构，相似性和差异有助于用户更好地分析社区和整体拓扑之间的关系，从而更深入地了解社区动态。

2.2 图结构分析

图挖掘涉及使用图论和网络分析等方法来探索，挖掘和理解图中的模式。

图形聚类。分析图中的社团结构是图挖掘中的重要任务之一。常用的方法包括子图划分，如 Louvain [9] 和 Infomap [42]。分析两个图之间的相似性可以通过计算图编辑距离来实现 [15]。该距离可以有效地帮助用户分析子图之间的结构相似性，也可以用于子图聚类。此外，图嵌

入是将图的结构信息转换成向量表示的另一个重要方法 [16,33,37]。通过图嵌入，可以将图映射到向量空间中，然后使用聚类算法（如 k 均值 [19]、谱聚类 [31] 等）。以聚类图向量，由此聚类相似的图。

图形对齐方式。当用户需要分析两个图中节点之间的对应关系时，图对齐 [34,39] 是一种有效的方法。通过图对齐，可以计算两个图之间的图相似度矩阵，得到两个图之间任意节点的相似度。基于此相似度矩阵，可以推断出相似节点之间的对应关系，从而进一步理解和分析两个图之间的结构和关系。

图形摘要。图表的描述性分析也很重要 [30]。例如，*VoG* 方法 [28] 是一种基于最大描述原理的大规模图分析方法。该方法从一个大图中提取最具描述性的子图结构，帮助用户理解其整体特征。另一方面，*Momo* 方法 [7] 侧重于分析和描述两个或多个图之间的相似性和差异性。该方法可以比较不同图之间的结构、节点属性、边关系等特征，帮助用户发现图之间的共性和差异。此外，*SSumM* 方法 [29] 生成稀疏汇总图，汇总图的信息以节省存储空间和分析时间。

图结构分析方法可以帮助用户发现重要信息，但其结果往往不直观。该方法采用相关图挖掘方法提取社区并识别社区的异同，设计有效的可视化技术帮助用户直观地查看和理解社区。

3 自适应 Motif 设计

3.1 方法概述

此部分对本文将要复现的工作进行概述，方法示意图如图 1 所示：

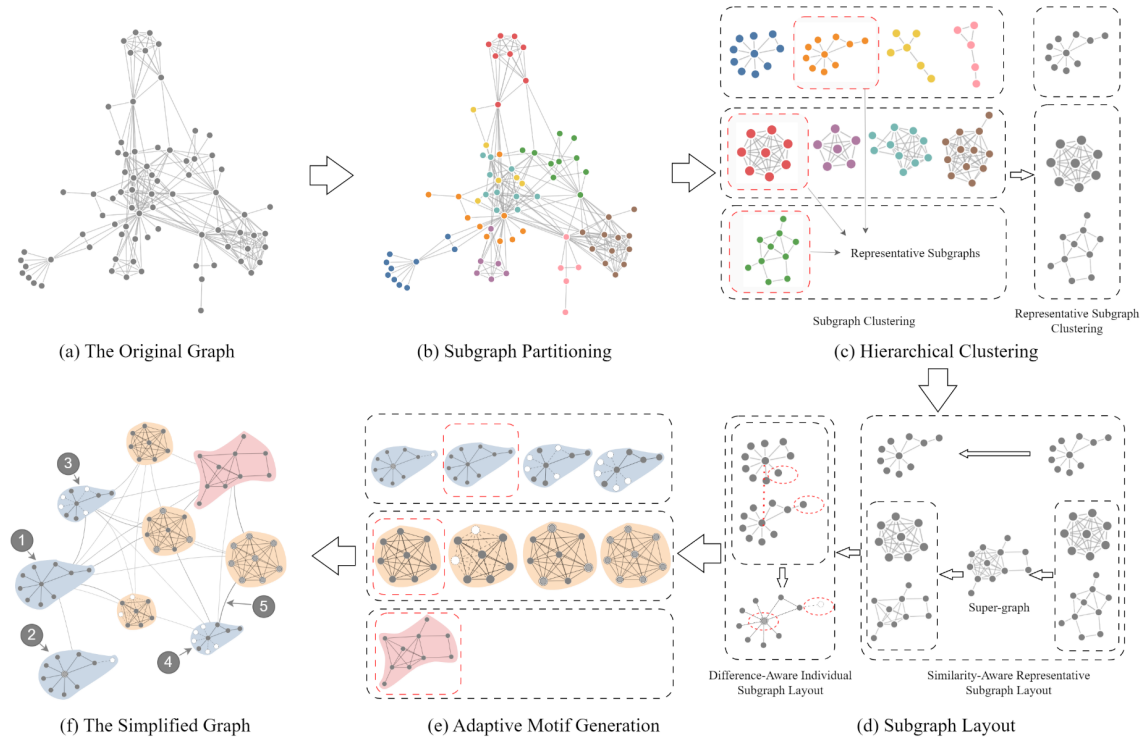


图 1. 方法示意图

AdaMotif 是一种新颖的图形简化方法，旨在提取和编码图形结构到自适应图案，从而简化图形，并帮助用户快速捕获图形社区结构和理解整体和局部图形结构。

图 1 说明了 *AdaMotif* 方法的框架。原始的力导向布局图有 77 个节点和 254 条边，如图 1 (a) 所示。该图被分成九个子图，每个子图在图 1 (b) 中用不同的颜色注释。这九个子图被聚类为三个类别，由左侧显示的灰色虚线框表示图 1 (c) 的一部分，并且每个类别被选择为具有代表性的子图，用红色虚线框突出显示。这三个代表性子图被聚集为两个类别，由图 1 (c) 右侧所示的灰色虚线框指示。对于具有两个图的代表性子图类别，合成一个超级图，以进一步在视觉上表示它们在布局中的相似性，如图 1 (d) 的右侧部分所示。在固定所有代表性子图的布局之后，将图 1 (c) 中的每个非代表性子图与其代表性子图进行比较，以便生成其布局。相似的节点被保留，而不同的节点用白色或灰色外环标记。图 1 (c) 中的每个子图的九个 *Motif* 被生成并在图 1 (e) 中示出。最后，根据各个子图中的节点数量缩放图案的大小，并且通过执行总体布局来生成最终结果（参见图 1 (f)）。在下面的小节中，将详细描述每个步骤。

3.2 子图划分

社区信息是一个关键的图结构。通过使用成熟的社区检测算法 [3]，可以自动识别社区子图，而无需事先设置社区的数量或大小。图 1 (b) 示出了图 1 (a) 的分割结果，其中九个子图具有不同的颜色。在图 1 (b) 中，虽然大多数社区都很容易识别，但黄色、橙子和青色社区交织在一起，这使得视觉分析具有挑战性，并突出了对图挖掘算法的需求。

3.3 层次聚类

本文的层次聚类包括两个步骤：首先，对子图进行聚类，然后，对于每个子图类别，识别代表性子图并进一步对这些代表性子图进行聚类。

子图聚类。在检测到社区子图之后，希望进一步清晰地呈现这些子图，并允许直观地观察它们之间的相似性。对于子图相似性，使用由 FEATHER [37] 计算的代表性向量来测量它，并使用亲和传播方法 [13] 来聚类子图。仿射传播与自动生成设计的目标非常一致，因为它不需要预先指定聚类的数量。如图 1 (c) 所示，九个子图被聚类为三个类别，每个类别由虚线框指示。可以观察到，在每个类别内的子图之间存在一定程度的结构相似性。

代表性子图聚类。对于每个子图类别，根据该类别的聚类中心计算代表子图。这些代表性的子图表现出较少的差异，从其他子图在同一类别，并用于后续的自适应 *Motif* 生成。此外，对这些有代表性的子图进行另一轮聚类。图 1 (c) 中的三个代表性子图进一步分为两类，并由灰色虚线框表示。在下一小节中，将根据这个聚类结果来计算代表性子图的布局，展示它们之间的局部结构相似性。

3.4 子图布局

自适应图案设计高度基于每个聚类的代表子图。自适应 *Motif* 被用来进一步表示原始图，因此，图的结构可以被有效地简化。然而，这种简化可能导致信息丢失。同时可以发现，同一聚类中的子图与同一聚类中的代表子图之间存在着一定的相似性和差异性。因此，通过表达这些相似性和设计上的差异。可以将子图布局分为两个组成部分：一个相似性感知的代

表子图布局被设计来说明代表子图之间的相似性，和一个差异感知的个体子图布局被设计来说明同一集群中的子图之间的差异。

相似性感知代表子图布局。在层次聚类步骤中，将具有代表性的子图分成簇，每个簇具有一定的相似性。需要提取相似之处并将其编码到布局中。在图匹配领域，最大公共子图方法可以用来准确地提取这种图的相似性。然而，这种方法是一个 NP 难问题 [20, 27]，并且仅适用于图相对较小的情况。因此，选择图形对齐方法 [34, 39] 来获得不准确的图形匹配结果。尽管存在一些错误，该方法具有成熟的算法和较低的时间复杂度，使其能够有效地检测近似图的相似性信息。

在 *AdaMotif* 方法中，采用 LREA [34] 方法作为图对齐的功能模块。它输出来自两个图的每对节点的相似度值。优先匹配节点对高节点度和高相似度值，而节点对低相似度值被忽略。由于图对齐算法一次只适合对齐两个图，对于具有多个代表子图的聚类，依靠合成的超图来实现对齐效果。

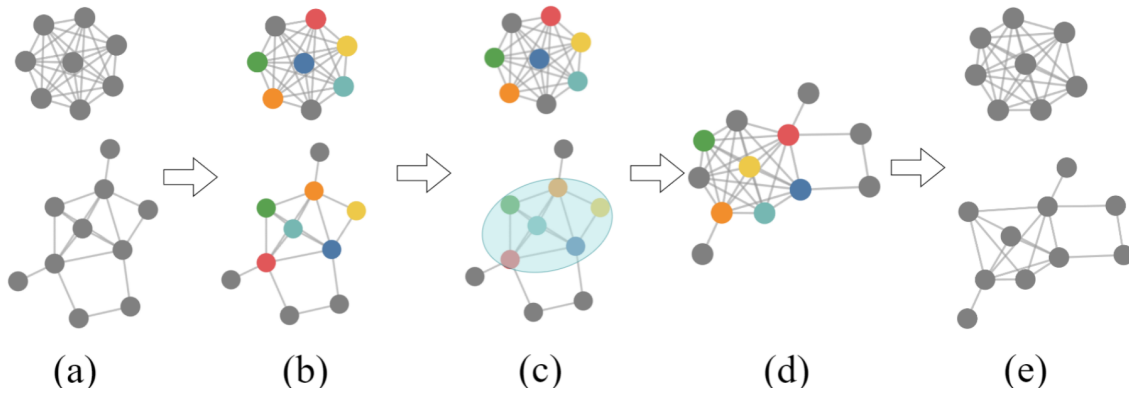


图 2. 相似性感知代表子图布局算法的示例

对于每个代表性子图聚类，如果只有一个图，则不合成超图（例如，图 1 (d) 顶部的群集）。如果存在多个图形（例如，图 1 (d) 底部的聚类），会选择其中一个图作为基础，并对齐另一个图以生成超级图。图 2 示出了生成超图的过程的示例。在图 2 (a) 中，两个代表性子图属于同一类别，表明它们之间具有一定程度的相似性。因此，进一步使用 LREA 方法来计算它们的对齐节点（即，相似的节点），并在图 2 (b) 中对它们进行着色。在上图和下图中具有相同颜色的节点是对齐的节点，而灰色节点是未对齐的节点。随后，在图 2 (c) 中，使用上图作为基础。移除下部图中的对齐的节点，并且然后基于剩余的未对齐的节点与移除的对齐的节点的连接将剩余的未对齐的节点连接到上部图中的对应的对齐的节点，从而产生图 2 (d) 中所示的超图。如果在该代表性子图类别内存在其他图，则超级图将随着其与其他图一一对齐而不断更新。接下来，将力导向节点链接图布局²应用于超级图，以获得所有节点的布局信息（见图 2 (d)）。然后，将所有代表性子图的节点布局与它们的超图的相应节点布局对齐，以说明它们的相似结构（见图 2 (e)）。如图 2 所示，从图 2 (a) 到图 2 (e) 的两个图的布局变换便于观察它们的相似性。因此，这个具有相似性意识的代表性的例子子图布局演示了一种显示图相似性的有效方式。

²<https://observablehq.com/@d3/force-directed-graph-component>

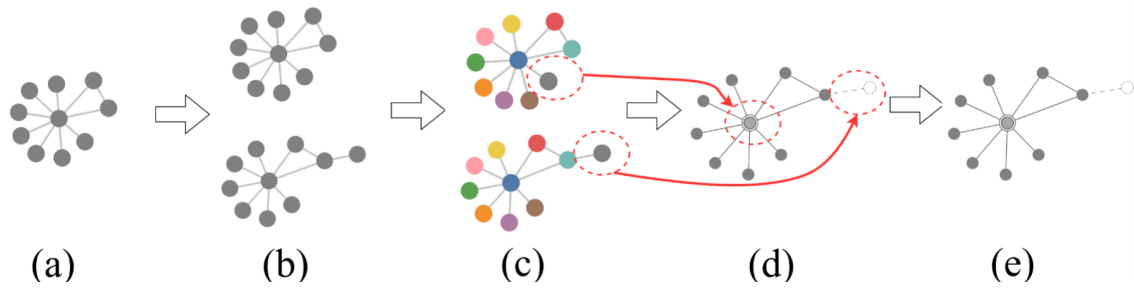


图 3. 差异感知的个体子图布局算法示例

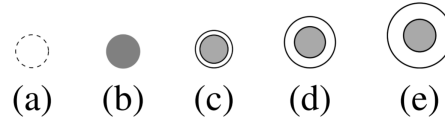


图 4. 差异感知的个体子图布局算法中的五种节点编码类型

差异感知单个子图布局。对于每个子图簇，使用代表子图来表示它们的相似性，但是它们也有差异。为了说明它们的差异，提出了一种差异感知的个体子图布局方法，图 3 中提供了一个例子。对于每个非代表性子图（例如，图 3 (a) 中的图），首先将其与代表性子图（例如，图 3 (b) 中的下图）使用图对齐来找到它们近似相同的部分（即，在图 3 (c) 中的两个图中具有相同颜色的节点）。未对准的部分（即，图 3 (c) 中的灰色节点）表示它们的差异。

由于自适应 *Motif* 是基于代表子图设计的，因此应该基于代表子图来说明差异。因此，基于代表子图，通过识别三种不同类型的节点：1) 属于代表子图但在子图中不具有对应对齐节点的未对齐节点（例如，图 3 (d) 中具有虚线边界的白色节点）；2) 与不属于代表性子图但在子图中具有对应对齐节点的未对齐节点连接的对齐节点（例如，图 3 (d) 中具有外环的灰色节点）；以及 3) 不与不在代表性子图中的任何未对准节点连接的对准节点（例如，图 3 (d) 中的正常灰色节点）。对于第一种类型的节点，连接到该节点的边也被设置为虚线。对于第二种类型的节点，外环的半径表示连接到此节点的未对齐节点的数量。半径越大，表示连接的未对齐节点越多。图 4 展示出了差异感知个体子图布局算法中的所有节点编码类型。图 4 (a) 示出了具有虚线边界的白色节点，图 4 (b) 示出了正常灰色节点，并且图 4 (c) - (e) 示出了 *AdaMotif* 方法中采用的具有外环的三种不同大小的灰色节点。所有带有虚线边界的白色节点和灰色节点的半径相同，灰色节点的外圈除外。只使用三种大小的半径作为外环来表示连接的未对齐节点的数量。图 4 (c) - (e) 中的外环半径对应于三个级别的节点数：小、中、大。这种编码方案可以帮助用户大致识别和区分不同级别的节点编号，而不会因为外圈太小而无法识别或太大而模糊附近的信息。可以观察到，使用图 3 (e) 的布局来表示图 3 (a) 使得更容易识别子图与其代表性子图之间的差异。

3.5 自适应 Motif 生成

自适应 *Motif* 是基于子图布局生成的。也就是说，基于相似性感知的代表子图布局来生成代表子图的 *Motif*，而基于差异性感知的个体来生成其他子图的 *Motif* 子图布局。*Motif* 的大小指示对应子图内的节点的数目。为了增强每个图案的可见性，为其添加了一个外部轮廓，并用颜色填充它们。不同的颜色表示不同的子图聚类。船体算法可用于提取每个子图的

轮廓 [21]。然而，由于每个图案的不规则形状，一些凸多边形不能非常紧密地包裹图案。使用凹面多边形可能更合适，这样用户就可以捕捉到图案的独特特征。因此，采用了支持凹多边形边界提取的 *alpha-shape* 算法 [12]。出于美观的目的，进一步利用多缓冲区算法³来扩大外轮廓，以便包围一定大小的节点。如图 1 (e) 所示，每个 *Motif* 是从图 1 (c) 中的相应子图生成的，并且使用图 1 (d) 中的子图布局表示来表示。在生成所有图案之后，接着执行力导向布局以优化整体布局。主题设计简化了图中的节点并减少了边。在 *AdaMotif* 设计中，原始图中节点之间的边被单个边连接 *Motif* 所取代，即使原始图中那些 *Motif* 内的节点之间存在多条边。通过使用边的灰度值来表示原始图中连接这两个 *Motif* 内的节点的边的数量。灰度值越暗，表示原始边的数量越多。另外，使用边捆绑技术 [25] 来增强 *AdaMotif* 方法整体的视觉效果。图 1 (f) 示出了图 1 (a) 的最终简化效果。该方法是完全自动化的，可以适应于具有不同类型子图的各种数据集的化简。

4 复现细节

4.1 与已有开源代码对比

在本文的基础上，尝试与其他布局方法进行对比，在使用本文方法的过程中在保留合并的超级图与聚类的图形数据结果，在 *Taurus* 中上传数据，然后选择不同布局方法，可以查看不同的布局结果。

4.2 实验环境搭建

我们使用具有 *Windows 10* 操作系统，*Intel Core i3- 12100 F 3.3GHz CPU*，*NVIDIA RTX 2060 12 GB* 显卡和 *16 GB* 内存，使用了 *python3.7* 与 *python3.9* 两个版本进行实验的具体运行。

4.3 界面分析与使用说明

在图 5 中是对于操作界面的示意图，可以处理和获取图社区相关数据后通过图 5 (a) 对齐数据进行保存，通过图 5 (b) 和 5 (c) 实现对于多边形社区边界的提取以及图形外轮廓外轮廓的扩大，在完成上述内容后，就可以得到最终的可视化布局界面。

4.4 创新点

本次复现内容提出了一个简化大型图形自适应图案设计框架。通过检测图中的社区，将其表示为不同的子图，接着对子图进行聚类，为每个类别计算代表性子图。在图对齐结果的基础上，采用相似度感知的代表子图布局算法和差异度感知的个体子图布局算法来确定子图布局。生成自适应 *Motif*，用这些 *Motif* 取代原来的子图。还通过 *Taurus* 在不同的方法上进行布局，进行多种方法的展示。另外对于子图布局有相似性感知与差异性感知两种。每个代表性子图簇，如果只有一个图，则不会合成超图。若有多个图，说明它们之间有一定程度的相似性，进一步使用 *LREA* 方法计算它们的对其节点，然后合并形成超级图。差异感知的

³<https://ww2.mathworks.cn/help/matlab/ref/polyshape.polybuffer.html>

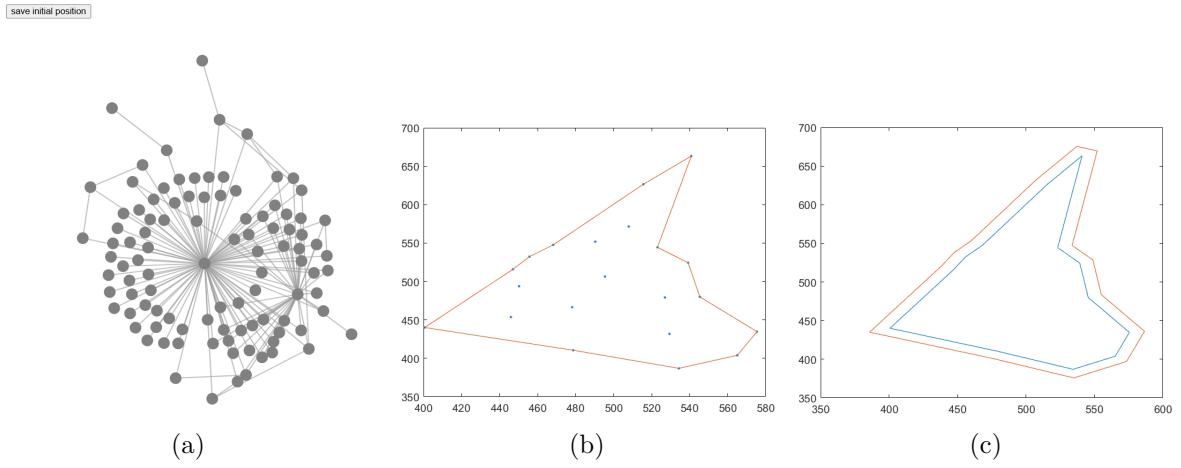


图 5. 操作界面示意

单个子图布局，将子图与代表性子图进行比较通过不同的视觉编码，说明同一集群中子图之间的差异。

5 实验结果分析

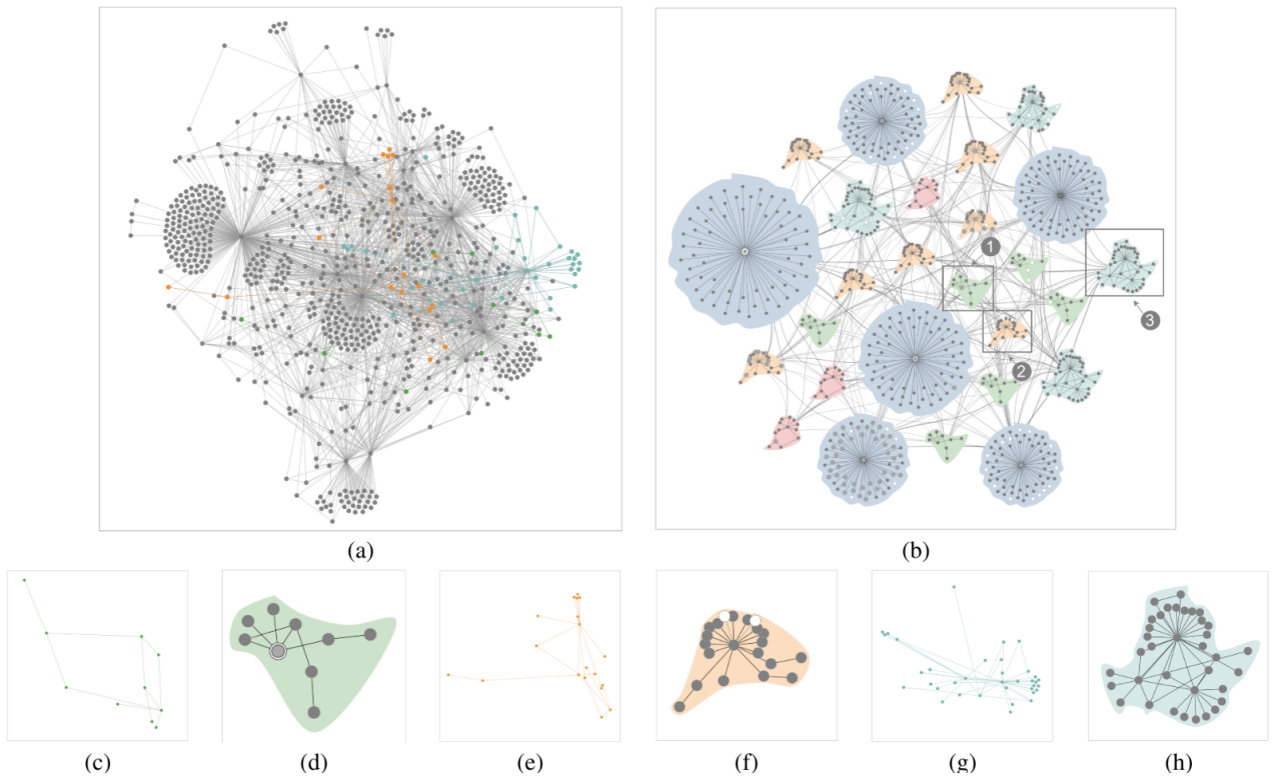


图 6. Cpan 结果图

图 6是在 *Cpan* 数据集上进行的案例研究，该数据集是一个拥有约 800 个节点和 2000 条边的协作网络。从左侧的原始图可以看出，只有具有大型星形结构的群落才能被清晰观察到，这里用灰色方框标出了其中一个。由于节点和边的密度较高，具有其他结构的群落很难识别。在右侧，我们的结果显示了具有大型星形结构和其他结构的群落，这些结构很容易识别。突

出显示的两个群落代表同一个群落。为了便于比较，我们将它们垂直排列。我们可以看到，我们的图案有效地捕捉到了这个群落的星形结构。在我们的结果中，许多在原始图中无法区分的结构都可以清晰地识别出来。例如，高亮显示的两个社区是同一个社区；为了便于比较，我们将它们水平对齐。您可以看到，在原始图中，这个群落的节点以红色高亮显示，很明显，这些节点相对分散，使得这个群落难以识别。不过，我们为这个社区设计的图案有效地捕捉到了它的结构。总的来说，我们的结果更清晰地呈现了所有的社区信息。

对所有不同规模的数据进行了统计分析，使用误差线表示性能。详情见图 7。总体而言，*AdaMotif* 在所有任务中的表现都优于原始图表。我们的方法在可理解性任务中获得了一致的较好结果。详情见图 8。对获得的统计结果进行了分析。对于每项任务，对收集到的数据进行了均值和标准误的测量分析。对于具有统计学显著性差异的测量，采用了以下检验方案：考虑到使用非标准化量表，必须首先确定数据分布的正态性。对每项任务的数据进行了 *Shapiro-Wilk* 检验，以验证正态分布的依从性。在正态分布的情况下，对具有显著差异的指标使用独立样本 *t* 检验。相反，对于不符合正态分布的数据，应用 *Wilcoxon* 符号秩检验。

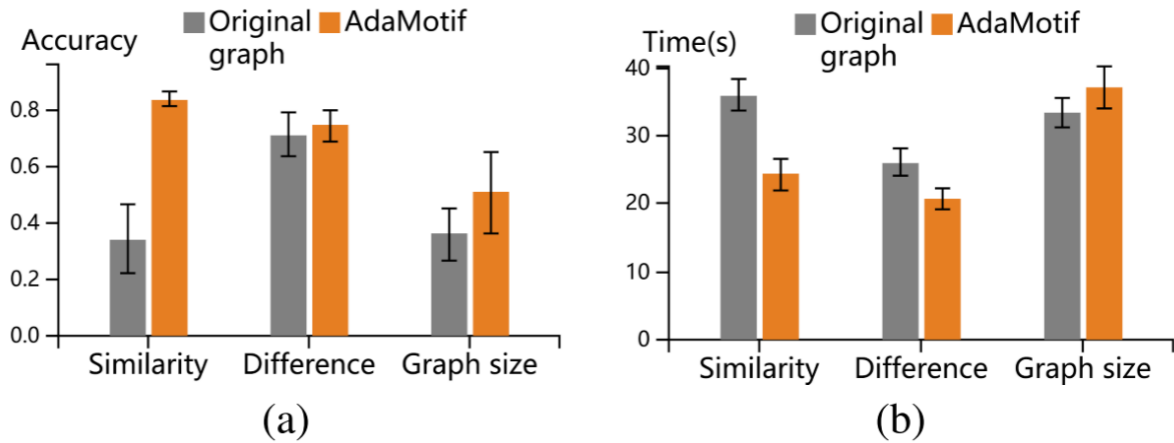


图 7. 原始图和 AdaMotif 在相似性、差异性和图大小上的平均值和标准误差

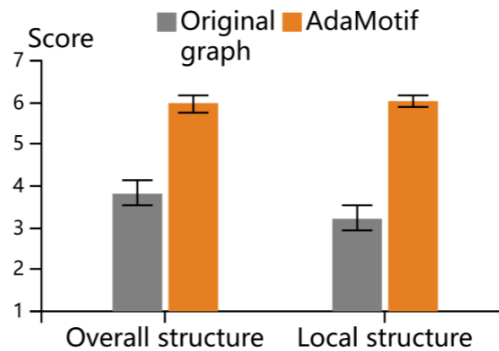


图 8. 原始图和 AdaMotif 在 7 point Likert 上的总体和局部结构的平均值和标准误差

6 总结与展望

本文中提出了一个新的框架，简化大型图形自适应 *Motif* 设计，它结合了图形可视化和图形挖掘技术的优势。首先检测图中的社区，表示为子图，然后聚类。为每个类别计算代表性子图。在图对齐结果的基础上，采用相似度感知的代表子图布局算法和差异度感知的个体子图布局算法来确定子图布局。最后，自适应图案的生成，这些图案取代原来的子图。*AdaMotif* 方法大大简化了原始图，使用户更容易分析和探索社区结构。但是 *AdaMotif* 方法目前只适用于无向图。在未来，计划研究其他类型的图的简化（例如，有向图和动态图）。

社区的相似性和差异反映在我们的图案是基于图聚类 and 图对齐算法利用。不同的图聚类算法可能会产生不同的结果，图对齐算法识别出的图异同点可能不准确。虽然我们知道最大公共子图方法会给出更精确的结果，但它是一个 NP 难问题，这严重限制了我们的方法在大型图上的效率。因此，我们选择了时间复杂度较低、不准确度较高的图对齐方法。在未来，我们计划探索是否有任何启发式的最大公共子图方法，以提高我们的方法的准确性。

对于社区结构清晰的图，该方法更加有效。当图不具有清晰的社区时，*AdaMotif* 方法在可视化大型图方面的优势可能会减弱。此外，这个方法对于大型图是有效的，需要注意的是，我们实现的算法仍然受到计算机内存和屏幕大小的限制。对于极大的图，*AdaMotif* 方法仍然会在计算时间和视觉混乱方面遇到可扩展性问题。在未来，将改进方法的可扩展性，以处理更大的图。一种可能性是使用 GPU 加速算法来处理超大图形，并使用多层显示器的设计方法来解决小屏幕的问题。

在图形简化领域，信息丢失是一个不可避免的挑战。该方法丢失了跨社区节点间的原始边信息。因此，计划在未来为结果中的简化边缘设计合适的编码方案。对于不同社区主题的布局，在当前的实现中使用了基本的力导向布局算法。将其他算法 [2, 40] 结合到我们的方法中以进一步增强我们的主题布局效果将是有趣的，这将作为未来的工作。

参考文献

- [1] Uri Alon. Network motifs: theory and experimental approaches. *Nature Reviews Genetics*, 8(6):450–461, 2007.
- [2] Daniel Archambault, Tamara Munzner, and David Auber. Topolayout: Multilevel graph layout by topological features. *IEEE Transactions on Visualization and Computer Graphics*, 13(2):305–317, 2007.
- [3] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(10):P10008, oct 2008.
- [4] Matthew Brehmer, Robert Kosara, and Carmen Hull. Generative design inspiration for glyphs with diatoms. *IEEE Transactions on Visualization and Computer Graphics*, 28(1):389–399, 2022.

- [5] Benjamin Cabrera, Björn Ross, Daniel Röchert, Felix Brünker, and Stefan Stieglitz. The influence of community structure on opinion expression: an agent-based model. *Journal of Business Economics*, 91(9):1331–1355, 2021.
- [6] Xiyuan Chen, Mark Heimann, Fatemeh Vahedian, and Danai Koutra. Cone-align: Consistent network alignment with proximity-preserving node embedding. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management, CIKM '20*, page 1985–1988, New York, NY, USA, 2020. Association for Computing Machinery.
- [7] Corinna Coupette and Jilles Vreeken. Graph similarity description: How are these graphs similar? In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, KDD '21*, page 185–195, New York, NY, USA, 2021. Association for Computing Machinery.
- [8] Yingan Cui, Xue Li, Junhuai Li, Huaijun Wang, and Xiaogang Chen. A survey of sampling method for social media embeddedness relationship. *ACM Comput. Surv.*, 55(4), November 2022.
- [9] Pasquale De Meo, Emilio Ferrara, Giacomo Fiumara, and Alessandro Provetti. Generalized louvain method for community detection in large networks. In *2011 11th International Conference on Intelligent Systems Design and Applications*, pages 88–93, 2011.
- [10] Nadezhda T Doncheva, Yassen Assenov, Francisco S Domingues, and Mario Albrecht. Topological analysis and interactive visualization of biological networks and protein structures. *Nature protocols*, 7(4):670—685, March 2012.
- [11] Cody Dunne and Ben Shneiderman. Motif simplification: improving network visualization readability with fan, connector, and clique glyphs. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '13*, page 3247–3256, New York, NY, USA, 2013. Association for Computing Machinery.
- [12] H. Edelsbrunner, D. Kirkpatrick, and R. Seidel. On the shape of a set of points in the plane. *IEEE Transactions on Information Theory*, 29(4):551–559, 1983.
- [13] Brendan J. Frey and Delbert Dueck. Clustering by passing messages between data points. *Science*, 315(5814):972–976, 2007.
- [14] Johannes Fuchs, Petra Isenberg, Anastasia Bezerianos, and Daniel Keim. A systematic review of experimental studies on data glyphs. *IEEE Transactions on Visualization and Computer Graphics*, 23(7):1863–1879, 2017.
- [15] Xinbo Gao, Bing Xiao, Dacheng Tao, and Xuelong Li. A survey of graph edit distance. *Pattern Anal. Appl.*, 13(1):113–129, February 2010.

- [16] Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 855–864, New York, NY, USA, 2016. Association for Computing Machinery.
- [17] Qingyu Guo, Fuzhen Zhuang, Chuan Qin, Hengshu Zhu, Xing Xie, Hui Xiong, and Qing He. A survey on knowledge graph-based recommender systems. *IEEE Transactions on Knowledge and Data Engineering*, 34(8):3549–3568, 2022.
- [18] Anchal Gupta and Rajesh Kumar Singh. Developing a framework for evaluating sustainability index for logistics service providers: graph theory matrix approach. *International Journal of Productivity and Performance Management*, 69(8):1627–1646, 2020.
- [19] J. A. Hartigan and M. A. Wong. Algorithm AS 136: A K-Means clustering algorithm. *Applied Statistics*, 28(1):100–108, 1979.
- [20] Juris Hartmanis. Computers and intractability: A guide to the theory of np-completeness (michael r. Garey and david s. Johnson). *Siam Review*, 24:90–91, 1982.
- [21] J. Heer and D. Boyd. Vizster: visualizing online social networks. In *IEEE Symposium on Information Visualization, 2005. INFOVIS 2005.*, pages 32–39, 2005.
- [22] Mark Heimann, Xiyuan Chen, Fatemeh Vahedian, and Danai Koutra. Refining network alignment to improve matched neighborhood consistency. *CoRR*, abs/2101.08808, 2021.
- [23] Judith Hermanns, Konstantinos Skitsas, Anton Tsitsulin, Marina Munkhoeva, Alexander Kyser, Simon Nielsen, Alexander M. Bronstein, Davide Mottin, and Panagiotis Karras. Grasp: Scalable graph alignment by spectral corresponding functions. *ACM Trans. Knowl. Discov. Data*, 17(4), February 2023.
- [24] Orland Hoeber. Information visualization for interactive information retrieval. In *Proceedings of the 2018 Conference on Human Information Interaction & Retrieval*, CHIIR '18, page 371–374, New York, NY, USA, 2018. Association for Computing Machinery.
- [25] Danny Holten and Jarke J. van Wijk. Force-Directed Edge Bundling for Graph Visualization. *Computer Graphics Forum*, 2009.
- [26] Bo Jiao, Xin Lu, Jingbo Xia, Brij Bhooshan Gupta, Lei Bao, and Qingshan Zhou. Hierarchical sampling for the visualization of large scale-free graphs. *IEEE Transactions on Visualization and Computer Graphics*, 29(12):5111–5123, 2023.
- [27] Viggo Kann. On the approximability of the maximum common subgraph problem. In *Symposium on Theoretical Aspects of Computer Science*, 1992.
- [28] Danai Koutra, U Kang, Jilles Vreeken, and Christos Faloutsos. Summarizing and understanding large graphs. *Statistical Analysis and Data Mining*, 8(3):183–202, 2015.

- [29] Kyuhan Lee, Hyeonsoo Jo, Jihoon Ko, Sungsu Lim, and Kijung Shin. Ssumm: Sparse summarization of massive graphs. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '20, page 144–154, New York, NY, USA, 2020. Association for Computing Machinery.
- [30] Yike Liu, Tara Safavi, Abhilash Dighe, and Danai Koutra. Graph summarization methods and applications: A survey. *ACM Comput. Surv.*, 51(3), June 2018.
- [31] Ulrike Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17:395–416, 01 2004.
- [32] Sadia Majeed, Muhammad Uzair, Usman Qamar, and Aftab Farooq. Social network analysis visualization tools: A comparative review. In *2020 IEEE 23rd International Multitopic Conference (INMIC)*, pages 1–6, 2020.
- [33] Annamalai Narayanan, Mahinthan Chandramohan, Rajasekar Venkatesan, Lihui Chen, Yang Liu, and Shantanu Jaiswal. graph2vec: Learning distributed representations of graphs. *ArXiv*, abs/1707.05005, 2017.
- [34] Huda Nassar, Nate Veldt, Shahin Mohammadi, Ananth Grama, and David F. Gleich. Low rank spectral network alignment. In *Proceedings of the 2018 World Wide Web Conference*, WWW '18, page 619–628, Republic and Canton of Geneva, CHE, 2018. International World Wide Web Conferences Steering Committee.
- [35] Peter Oliver, Eugene Zhang, and Yue Zhang. Scalable hypergraph visualization. *IEEE Transactions on Visualization and Computer Graphics*, 30(1):595–605, 2024.
- [36] Manish Purohit, B. Aditya Prakash, Chanhyun Kang, Yao Zhang, and V.S. Subrahmanian. Fast influence-based coarsening for large networks. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, page 1296–1305, New York, NY, USA, 2014. Association for Computing Machinery.
- [37] Benedek Rozemberczki and Rik Sarkar. Characteristic functions on graphs: Birds of a feather, from statistical descriptors to parametric models. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, CIKM '20, page 1325–1334, New York, NY, USA, 2020. Association for Computing Machinery.
- [38] Shazia Tabassum, Fabíola Souza F. Pereira, Sofia da Silva Fernandes, and João Gama. Social network analysis: An overview. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 8, 2018.
- [39] Huynh Thanh Trung, Nguyen Thanh Toan, Tong Van Vinh, Hoang Thanh Dat, Duong Chi Thang, Nguyen Quoc Viet Hung, and Abdul Sattar. A comparative study on network alignment techniques. *Expert Systems with Applications*, 140:112883, 2020.

- [40] Mingliang Xue, Zhi Wang, Fahai Zhong, Yong Wang, Mingliang Xu, Oliver Deussen, and Yunhai Wang. Taurus: Towards a unified force representation and universal solver for graph layout. *IEEE Transactions on Visualization and Computer Graphics*, 29(1):886–895, 2023.
- [41] Lu Ying, Xinhuan Shu, Dazhen Deng, Yuchen Yang, Tan Tang, Lingyun Yu, and Yingcai Wu. Metaglyph: Automatic generation of metaphoric glyph-based visualization. *IEEE Transactions on Visualization and Computer Graphics*, 29(1):331–341, 2023.
- [42] Jianping Zeng and Hongfeng Yu. A distributed infomap algorithm for scalable and high-quality community detection. In *Proceedings of the 47th International Conference on Parallel Processing*, ICPP ’18, New York, NY, USA, 2018. Association for Computing Machinery.
- [43] Ying Zhao, Haojin Jiang, Qi’an Chen, Yaqi Qin, Huixuan Xie, Yitao Wu, Shixia Liu, Zhiguang Zhou, Jiazhi Xia, and Fangfang Zhou. Preserving minority structures in graph sampling. *IEEE Transactions on Visualization and Computer Graphics*, 27(2):1698–1708, 2021.
- [44] Sijin Zhou, Xinyi Dai, Haokun Chen, Weinan Zhang, Kan Ren, Ruiming Tang, Xiuqiang He, and Yong Yu. Interactive recommender system via knowledge graph-enhanced reinforcement learning. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR ’20, page 179–188, New York, NY, USA, 2020. Association for Computing Machinery.
- [45] Youjia Zhou, Archit Rathore, Emilie Purvine, and Bei Wang. Topological simplifications of hypergraphs. *IEEE Transactions on Visualization and Computer Graphics*, 29(7):3209–3225, 2023.
- [46] Zhiguang Zhou, Chen Shi, Xilong Shen, Lihong Cai, Haoxuan Wang, Yuhua Liu, Ying Zhao, and Wei Chen. Context-aware sampling of large networks via graph representation learning. *IEEE Transactions on Visualization and Computer Graphics*, 27(2):1709–1719, 2021.