

# 让图表发光：利用文本到图像生成模型嵌入语义上下文到图表中

张镇耀

## Abstract

图形可视化将数据与语义上下文融合在视觉表示中从而传递复杂信息。研究着眼于简化图形可视化工具，但主流方法通常依赖于专用语料库，可能影响数据完整性。本研究介绍了 ChartSpark，一个基于文本输入中传达的语义上下文和嵌入在简单图表中的数据信息的图形可视化的系统。我们还开发了交互式界面，包括文本分析、编辑和评估模块，用户可生成、修改和评估图形可视化。实验证明了工具的可用性，并讨论了文本到图像生成模型与交互式界面相结合在可视化设计中的潜力。

关键词：图形可视化，生成模型，创作工具。

## 1 引言

可视化技术因其清晰且生动地展示复杂的数据的能力而被普遍应用。图形可视化技术能通过将语义上下文嵌入到图表中增强数据中隐藏的视觉表示 [10, 11, 23]。图形可视化有诸如提高长期记忆等优势 [2, 8, 9, 18]。其在广告，教育等领域有美学和实用价值 [1, 22, 3, 12, 17]。

近期的研究允许用户通过选择图形元素，然后将数据绑定在上面，从而创建一个可视化的图表。这些方法一般有两个阶段，从大图像语料库中进行检索，随后在交互式界面进行编辑微调。Infomages 和 Vistylist 采用了这些方法。后来语言模型开始流行，人们深入研究文本生成图表 [7, 14, 21, 24]，弥补可视化与自然语言之间的鸿沟。

这些方法存在一些共同的问题，比如检索需要大规模且高质量的图像语料库，这在构建上是具有挑战性的。在这个方向的研究中，往往会创建新的数据集，导致存在不符合彼此需求的各种数据集 [7, 15, 16, 19]。其次，编辑图像以将其与数据绑定是一个容易出错的任务。此外，当前文本的引导方法还受到诸如与比例相关的统计信息限制。

我们提出一种新颖的图形可视化工具用来缓解这些限制，其名为 CharkSpark。这个工具基于文本到图像生成模型，将语义上下文嵌入图表，同时保留了重要的视觉属性，比如趋势。我们从图表的标题提取目标可视化对象，由对象及用户提供的相应描述来指导生成过程。这个工具有三个优势，其一是消除了搜索繁琐性和可能搜索不到的情形，其二是更加自然地将语义上下文嵌入到图表中，其三是涵盖了更多视觉元素并支持灵活的文本指令。我们的贡献主要有如下三点：

- 我们基于生成模型创建图形可视化，该模型可以由语义上下文和结构信息进行条件约束。
- 我们构建了一个界面，包括文本分析器、编辑模块和评估模块，以协助用户进行创作过程。
- 案例研究和专家访谈证明了我们方法的有效性。

## 2 相关工作

### 2.1 图形可视化

图形可视化能够以引人入胜的方式呈现复杂数据，具有更强的美学吸引力，可以使主题形象化从而增强参与度，提高信息吸收和记忆力。Borkin 等人的“超越记忆”研究表明 [22]，正确使用的符号不会干扰理解，并且可以提高识别能力。一些研究专注于拓展设计空间 [4, 19]，提出了各种设计维度作为实践的指导。近期，更多工作致力于研究结合语义背景和数据嵌入类型。我们的工作旨在从大量图形可视化语料库中对各种嵌入类型进行分类和组织，并支持灵活的生成。

### 2.2 可视化创作工具

先前的研究 [13, 26, 27] 侧重于使用适当的视觉编码通道将视觉元素映射到数据。为了简化设计视觉元素的过程，许多方法实现了一个检索和编辑的流水线，从一个庞大的资源库中获取灵感。基于检索有两种设计类型：基于内容的设计和基于示例的设计。基于内容的设计 [6, 28] 包括搜索与原始图表共享语义背景的图像，并将其视觉表示与图表合并。基于示例的设计 [5, 20, 23, 25] 侧重于检索一个精心制作的图表作为参考，并模拟其样式以创建新的内容。尽管这些方法表面上看起来简单，但仍需要图形可视化设计专业知识，这对初学者用户可能是具有挑战性的。在这项工作中，我们开发了一个创作工具，简化了创建过程，减轻了检索适当图像和在手动编辑视觉设计时潜在错误的挑战。

### 2.3 文本引导图像生成

文本引导图像生成文本到图像扩散模型已经展示出在生成高质量图像方面取得了巨大成功，超越了先前的主流 GAN 模型和自回归模型。我们提出了一个通用生成流程，包括对几种不同类型图表的全面嵌入类型和生成方法。这个流程提供了一个端到端的生成过程，使用户从手动设计初始图表的负担中解脱出来。此外，我们集成了对生成图表的修改和评估，以提高性能。

## 3 本文方法

### 3.1 概述

提出的 ChartSpark 框架的工作流程包括三个主要阶段：(1) 在初始阶段，数据特征被可视化，并从原始数据中提取语义上下文，为用户提供视觉预览和主题主题，以增强对数据的理解。(2) 随后，用户使用以提示为驱动的生成来获取视觉元素，以有条件或无条件的方式将语义上下文嵌入到前景或背景中。(3) 最终，评估模块提供了一个建议机制，指示数据的扭曲程度。

### 3.2 特征提取

为了向用户提供其数据的预览，我们分别提取数据特征和语义上下文。在数据特征提取中，我们提供多种图表类型供用户选择进行可视化。为了提取语义上下文，我们采用了两步方法，即关键词提取和相关词检索。最初，我们使用了 MPNet，这是一个带有句子转换器的预训练模型，从原始数据中提取标题中的关键词。此外，我们提供相关词来激发用户的创造力，特别是那些具有有限设计专业知识的用户。我们使用 Word2Vec 估算单词相似性并将单词转换为向量。

### 3.3 无条件生成

文本引导图像生成我们设计了一种生成方法,采用了无条件和有条件的方法。这些方法之间的根本区别在于图表信息是否被纳入生成过程。无条件生成我们基于 Frozen Latent Diffusion Model (LDM) 开发了我们的框架。为了实现文本引导生成, LDM 通过交叉注意力机制增强了底层 UNet 骨干, 便于集成多模态引导。无条件生成可以生成用于修饰图表的视觉元素。我们的目标是从图像中提取与语义上下文  $P_{obj}$  相关的视觉元素。过程如图 1 所示。

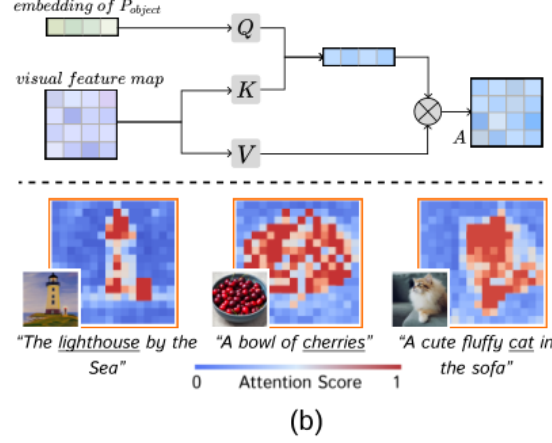


Figure 1: 将图像和文本输入纳入注意力地图的内部机制。

为了实现这一点, 我们使用对象  $P_{obj}$  和  $I_g$  之间的交叉注意力来定位目标区域, 然后去除背景以获得  $I_{obj}$ 。我们从自动编码器和  $P_{obj}$  的嵌入中获得生成图像的视觉特征图  $V$ 。接下来, 我们使用线性投影将它们转换为  $Q$  和  $K$ 。然后, 我们将  $Q$  和  $K$  相乘以获得注意力分数, 随后将其与  $V$  相乘以生成最终的注意力图。总体而言, 该过程可以描述如下:

$$A(Q, K, V) = \text{Softmax}\left(\frac{QK^T}{\sqrt{n}}\right) \cdot V \quad (1)$$

其中  $d$  表示  $Q$  和  $K$  的潜在投影维度,  $\text{Softmax}$  函数用于归一化注意力分数。因此, 我们可以通过比较像素差异从混乱的背景中提取感兴趣的对象。为了实现这一目标, 首先计算阈值来区分对象和背景, 得到一个掩码。接下来, 在  $I_g$  中对应位置进行逐像素比较, 得到一个粗略的对象区域, 表示为  $I_{obj}$ 。最后, 为了获得更精细的结果, 我们利用 ISNet, 这是一种先进的分割神经网络, 消除冗余信息。该过程可以通过以下方程描述:

$$M = I[A_{ij} > \frac{\sum A_{ij}}{N^2}] \quad (2)$$

$$I_{obj} = f_{upsample}(M) \odot I_g \quad (3)$$

$$I_{obj}' = R(I_{obj}) \quad (4)$$

其中,  $I[\cdot]$  是矩阵上的逐元素指示函数,  $N^2$  表示注意力图  $A$  中像素的总数,  $M$  被计算为一个矩阵, 在对象的位置为 1, 其他地方为 0。由于  $M$  具有与  $A$  相同的维度, 我们使用上采样技术将其形状调整为与  $I_g$  匹配。 $R$  表示冗余信息去除操作。

### 3.4 有条件生成

与无条件生成相比，有条件生成涉及整合图表图像，使生成的视觉元素符合数据信息，有两个主要的挑战需要解决：1) 增强生成多样性。2) 集成语义上下文和图表。

语义上下文需要整合到条形图、折线图、饼图和散点图的矩形、线条、扇形和气泡中。最初，我们通过各种操作（包括高斯模糊、动态模糊和图像变形）随机增强  $I_c$ 。增强模块  $aug$  基于增强图表元素形状多样性的原则建立，同时保持数据完整性。然后，我们从生成过程中获得关于  $P_{object}$  的注意力图  $A$ 。为了将来自  $I_c$  的数据信息注入到注意力图中，我们将  $I_c$  用作掩码，确保注意力图具有与  $I_c$  中的元素相同的形状。为了最大化融合图像  $I_{fuse}$ ，包含尽可能多的语义上下文，我们采用两个常见的仿射变换，即缩放和旋转。优化函数可以表示为：

$$I_{fuse} = \max_{\theta, s} [aug(I_c) \odot \phi(f_{upsample}(A), \theta, s)] \quad (5)$$

其中， $\phi$  是由缩放参数  $s$  和旋转参数  $\theta$  参数化的仿射变换。最终，我们将  $I_{fuse}$  作为输入，将语义上下文和图表信息整合在一起，得到  $I_g$ 。

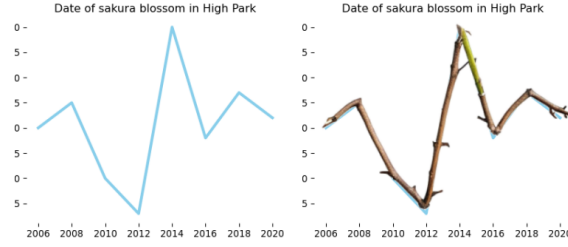


Figure 2: 条件生成前景。

背景不仅作为语义上下文的容器，还作为传达数据信息的图表的一部分。为了确保图表特征与背景组件之间的无缝整合，我们设计了一种新颖的融合策略，结合了颜色和上下文语义。在实践中，我们基于注意力图提取主导颜色，然后合成一个注入了  $I_c$  的颜色掩码。得到的颜色掩码与语义信息串联在一起，得到  $I_{fuse}$ 。然后，我们将  $I_{fuse}$  注入生成过程中以实现重建，得到  $I_g$ 。过程如图 4 所示。

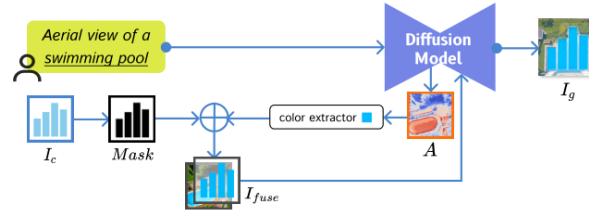


Figure 3: 条件生成用于背景。

### 3.5 评估

评估模块设计用于考虑数据的有条件生成。鉴于每个图表都采用不同的可视通道来编码数据，我们调整我们的方法以确保对每种图表类型进行可靠的评估。对于条形图，我们主要关注高度作

为失真的指标。对于折线图，趋势的描绘至关重要。对于饼图，我们测量每个扇区的角度。对于散点图，我们估计每个点的大小。我们从生成的元素中提取中心轴，用一个框滤波器平滑像素值，并使用一个窗口沿着图表  $I_c$  和生成的元素  $I_g$  滑动。在每个窗口中，我们将计算平均像素值以获取得分，如下所示：

$$S_i = 1 - \frac{|W_{I_c}^i - W_{I_g}^i|}{255} \quad (6)$$

得分越高，两个窗口之间的相似性越大。此外，我们可视化该得分的窗口，用一个红色的正方形表示失真区域。通过对沿着折线的每个窗口得分取平均，我们得到全局分数。

## 4 复现细节

### 4.1 与已有开源代码对比

源代码的地址连接在：<https://github.com/serendipitysx/chartspark>，包含前后端的工程文件。在本次复现中，尝试了有条件前景生成的另外一种方法。我采用了 stable diffusion 作为扩散模型，同时加入了 controlnet 来控制有条件前景的生成。我将数据表的折线方框或者饼状图形状作为蒙版输入到扩散模型中，随后使用 controlnet 提取蒙版边缘。之后，以蒙版边缘和提示词作为条件输入，交由扩散模型生成论文所需要的有条件前景。最后，使用蒙版提取出生成的有条件前景图相关部分，与一张全透明的图片合成，导入到 ChatSpark 中进行编辑操作。

基于 controlnet 生成的有条件背景方法相较于原文方法的优点在于，能够更加精准可控地让生成的图片边缘形状与数据表的折线，方框或饼状图形状相同。另外，还可以通过修改拟合权重，使得图片的生成更加自然，比如需要生成的物体边缘比较圆滑，而蒙版边缘比较生硬时我们可以通过减小拟合权重来生成边缘较为圆滑的有条件前景图。

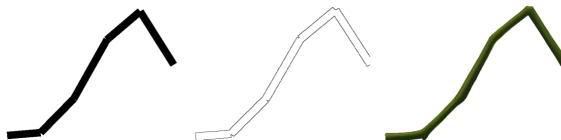


Figure 4: 基于 controlnet 生成有条件前景。

### 4.2 实验环境搭建

源代码后端使用 python 语言编写，使用 flask 框架搭建 web 应用。Python 设计简洁，拥有大量的库和框架，如 NumPy、Pandas、TensorFlow 等，可以加速开发过程。Python 可用于多种用途，包括 Web 开发、数据分析、人工智能、科学计算等领域。Flask 是一个轻量级的 Python Web 框架，它设计简单、灵活，适用于快速构建 Web 应用程序。这个框架是由 Armin Ronacher 开发的，基于 Werkzeug 工具箱和 Jinja2 模板引擎。Flask 使用 Werkzeug 库处理 HTTP 请求和响应，提供了一种灵活的 URL 路由方式。

源代码前端是运行在 node.js 上的程序。Node.js 在前端开发中扮演着多重角色，能够供了各种构建工具和打包器（比如 Webpack、Parcel 等），用于处理前端资源（如 JavaScript、CSS、图像等），使其在生产环境中加载更高效。也可用于构建服务器和编写后端逻辑。在前端开发中，开



发者可以使用 Node.js 构建轻量级的服务器或者 API 服务，也方便进行前后端集成与通信。源代码前端使用 vue 来构建 web 界面，使用 axios 进行前后端通信交互。

### 4.3 实验运行过程

前端和后端同时运行后，需要保持网络能够正常连接。在运行过程中，仍需要从网络中获取相应的资源。程序运行初始页面如图 5 所示。

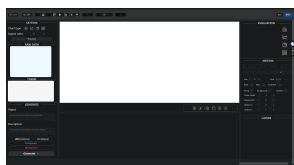


Figure 5: 程序初始界面



Figure 6: 生成数据图表

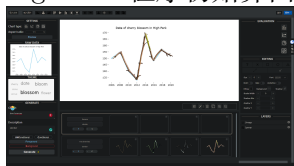


Figure 7: 条件生成图片



Figure 8: 无条件生成图片

首先点击左上角“导入文件”，这里的文件指的 json 文件，里面记录了绘制图表的基本数据，比如樱花盛开的花期。导入之后，选择绘制图表类型，选择折线图，然后点击 preview，系统将根据 json 文件记载的数据生成相应的折线图，同时使用词云展示主题（图 6）。接下来到生成环节，我们可以先生成樱花树的树枝，拟合折线图的线（图 7）。随后我们生成樱花，拟合折线图的节点（图 8）所示。最后，我们在右侧调整一些参数，就可以把图片保存到本地。

### 4.4 实验遇到的问题

在本地部署时，发现 Flask==2.2.2 下载后，会出现“ImportError: 无法从'werkzeug'导入名称'url\_encode'”的问题。通过查阅资料，发现是版本的问题，于是在本地部署时采用“pip install flask=3.0.0”，将 flask 和 werkzeug 更新到最新版本。

在本地部署时，会出现“Error: Could not locate a Flask application. You did not provide the “FLASK\_APP” environment variable”的问题。这是因为 flask 并没有找到文件名为 app.py 或者 wsgi.py 的文件，同时也没有使用 FLASK\_APP 命令更改需要载入的应用。此时需要在命令行输入“set FLASK\_APP=main”，同时修改主程序名称为 app。

另外，源代码中还会出现字体 fonts 找不到，图片找不到的问题，在代码中修改文件路径，同时手动不全相应的文件夹。

当程序成功运行时，我们会发现程序通过 flask 运行在端口号 5000 上，而源代码前后端交互时的端口号是 88，因此需要将端口号修改为 5000。

## 5 实验结果

为了对 ChartSpark 与之前的工具（包括 DataQuilt [60] 和 Infomages [11]）的性能进行全面比较，我们邀请了四位在信息设计领域拥有丰富经验的设计师和专家来评估这些工具。我们建立了六个指标，全面涵盖了创建图形可视化以详细评估每个工具的过程。完成评估过程后，每位参与者基于六个评估指标为每个工具提供了评分。结果如图 9 所示：

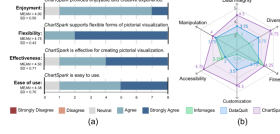


Figure 9: 用户研究和专家访谈结果。

实验结果表明，ChartSpark 消除了搜索图像的需要，并允许生成尽可能多的可视化，在对语义的忠实度，表现力上表现优秀，同时系统可用性也得到用户的认可。

## 6 总结

在这篇论文中，我们提出了一个名为 ChartSpark 的新型框架，用于创建图形可视化。该框架采用了一个文本到图像的生成模型，将语义上下文和图表信息整合在一起。ChartSpark 灵活且兼容无条件和有条件的方法，以融合语义上下文和图表。它还具有多功能性，可应用于前景和背景可视化。此外，我们开发了一个用户界面，集成了特征提取、生成和评估，以促进创建过程。为了评估 ChartSpark 的有效性，我们展示了涵盖四种主要形式的图形可视化的几个案例。来自用户研究和专家访谈的反馈证明了该框架的有效性。

## References

- [1] BATEMAN, S., MANDRYK, R. L., GUTWIN, C., GENEST, A., MCDINE, D., AND BROOKS, C. Useful junk? the effects of visual embellishment on comprehension and memorability of charts. In *Proceedings of the SIGCHI conference on human factors in computing systems* (2010), pp. 2573–2582.
- [2] BORGO, R., ABDUL-RAHMAN, A., MOHAMED, F., GRANT, P. W., REPPA, I., FLORIDI, L., AND CHEN, M. An empirical study on using visual embellishments in visualization. *IEEE Transactions on Visualization and Computer Graphics* 18, 12 (2012), 2759–2768.
- [3] BORKIN, M. A., VO, A. A., BYLINSKII, Z., ISOLA, P., SUNKAVALLI, S., OLIVA, A., AND PFISTER, H. What makes a visualization memorable? *IEEE transactions on visualization and computer graphics* 19, 12 (2013), 2306–2315.
- [4] BOY, J., PANDEY, A. V., EMERSON, J., SATTERTHWAITE, M., NOV, O., AND BERTINI, E. Showing people behind data: Does anthropomorphizing visualizations elicit more empathy for human rights data? In *Proceedings of the 2017 CHI conference on human factors in computing systems* (2017), pp. 5462–5474.

- [5] CHEN, Z., WANG, Y., WANG, Q., WANG, Y., AND QU, H. Towards automated infographic design: Deep learning-based auto-extraction of extensible timeline. *IEEE transactions on visualization and computer graphics* 26, 1 (2019), 917–926.
- [6] COELHO, D., AND MUELLER, K. Infomages: Embedding data into thematic images. In *Computer Graphics Forum* (2020), vol. 39, Wiley Online Library, pp. 593–606.
- [7] CUI, W., ZHANG, X., WANG, Y., HUANG, H., CHEN, B., FANG, L., ZHANG, H., LOU, J.-G., AND ZHANG, D. Text-to-viz: Automatic generation of infographics from proportion-related natural language statements. *IEEE transactions on visualization and computer graphics* 26, 1 (2019), 906–916.
- [8] HAROZ, S., KOSARA, R., AND FRANCONERI, S. L. Isotype visualization: Working memory, performance, and engagement with pictographs. In *Proceedings of the 33rd annual ACM conference on human factors in computing systems* (2015), pp. 1191–1200.
- [9] HARRISON, L., REINECKE, K., AND CHANG, R. Infographic aesthetics: Designing for the first impression. In *Proceedings of the 33rd Annual ACM conference on human factors in computing systems* (2015), pp. 1187–1190.
- [10] HARTMANN, F. Visualizing social facts: Otto neurath’s isotype project. In *European modernism and the information society*. Routledge, 2017, pp. 279–293.
- [11] HOLMES, N. *Joyful Infographics: A Friendly, Human Approach to Data*. CRC Press, 2022.
- [12] HULLMAN, J., ADAR, E., AND SHAH, P. Benefitting infovis with visual difficulties. *IEEE Transactions on Visualization and Computer Graphics* 17, 12 (2011), 2213–2222.
- [13] KIM, N. W., SCHWEICKART, E., LIU, Z., DONTCHEVA, M., LI, W., POPOVIC, J., AND PFISTER, H. Data-driven guides: Supporting expressive design for information graphics. *IEEE transactions on visualization and computer graphics* 23, 1 (2016), 491–500.
- [14] LAI, C., LIN, Z., JIANG, R., HAN, Y., LIU, C., AND YUAN, X. Automatic annotation synchronizing with textual description for visualization. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (2020), pp. 1–13.
- [15] LAN, X., SHI, Y., ZHANG, Y., AND CAO, N. Smile or scowl? looking at infographic design through the affective lens. *IEEE Transactions on Visualization and Computer Graphics* 27, 6 (2021), 2796–2807.
- [16] LU, M., WANG, C., LANIR, J., ZHAO, N., PFISTER, H., COHEN-OR, D., AND HUANG, H. Exploring visual information flows in infographics. In *Proceedings of the 2020 CHI conference on human factors in computing systems* (2020), pp. 1–12.
- [17] MOERE, A. V., AND PURCHASE, H. On the role of design in information visualization. *Information Visualization* 10, 4 (2011), 356–371.



- [18] MOERE, A. V., TOMITSCH, M., WIMMER, C., CHRISTOPH, B., AND GRECHENIG, T. Evaluating the effect of style in information visualization. *IEEE transactions on visualization and computer graphics* 18, 12 (2012), 2739–2748.
- [19] MORAIS, L., JANSEN, Y., ANDRADE, N., AND DRAGICEVIC, P. Showing data about people: A design space of anthropographics. *IEEE Transactions on Visualization and Computer Graphics* 28, 3 (2020), 1661–1679.
- [20] QIAN, C., SUN, S., CUI, W., LOU, J.-G., ZHANG, H., AND ZHANG, D. Retrieve-then-adapt: Example-based automatic generation for proportion-related infographics. *IEEE Transactions on Visualization and Computer Graphics* 27, 2 (2020), 443–452.
- [21] RASHID, M. M., JAHAN, H. K., HUZZAT, A., RAHUL, R. A., ZAKIR, T. B., MEEM, F., MUKTA, M. S. H., AND SHATABDA, S. Text2chart: A multi-staged chart generator from natural language text. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining* (2022), Springer, pp. 3–16.
- [22] RECALL, E. R. Beyond memorability: Visualization recognition and recall. *IEEE TRANSACTIONS ON VISUALIZATION AND COMPUTER GRAPHICS* 22, 1 (2016).
- [23] SHI, Y., LIU, P., CHEN, S., SUN, M., AND CAO, N. Supporting expressive and faithful pictorial visualization design with visual style transfer. *IEEE Transactions on Visualization and Computer Graphics* 29, 1 (2022), 236–246.
- [24] WANG, Y., HOU, Z., SHEN, L., WU, T., WANG, J., HUANG, H., ZHANG, H., AND ZHANG, D. Towards natural language-based visualization authoring. *IEEE Transactions on Visualization and Computer Graphics* 29, 1 (2022), 1222–1232.
- [25] WANG, Y., SUN, Z., ZHANG, H., CUI, W., XU, K., MA, X., AND ZHANG, D. Datashot: Automatic generation of fact sheets from tabular data. *IEEE transactions on visualization and computer graphics* 26, 1 (2019), 895–905.
- [26] WANG, Y., ZHANG, H., HUANG, H., CHEN, X., YIN, Q., HOU, Z., ZHANG, D., LUO, Q., AND QU, H. Infonice: Easy creation of information graphics. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (2018), pp. 1–12.
- [27] XIA, H., HENRY RICHE, N., CHEVALIER, F., DE ARAUJO, B., AND WIGDOR, D. Dataink: Direct and creative data-oriented drawing. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (2018), pp. 1–13.
- [28] ZHANG, J. E., SULTANUM, N., BEZERIANOS, A., AND CHEVALIER, F. Dataquilt: Extracting visual elements from images to craft pictorial visualizations. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (2020), pp. 1–13.