# 读后感

读了老师发的三篇论文，其中我重点读了老师写的那篇，我大致知道都在说什么，但都不是特别理解。在老师这个作者面前，我只能发表一下我的浅显感悟，实属班门弄斧，有说错的地方还望老师笑纳。

第一篇是有关大型集群上的简化数据处理方面的，对应的编程模型和相关实现称为MapReduce。MapReduce是一个用于处理和生成大型数据集的编程模型和相关实现。用户指定处理键/值对的map两数来生成一组中间键/值对，以及合并与同一中间键相关的所有中间值的reduce函数。该模型可以表达许多现实世界中的任务。他可以使得没有任何并行和分布式系统经验的程序员可以轻松地利用大型分布式系统的资源。在过去的五年中,作者在谷歌和其他许多实现了数以百计的专用计算这一过程大量的原始数据,大多数这样的计算在概念上是简单的。然而，输入数据通常很大，为了在合理的时间内完成计算，必须将计算分布到数百或数千台机器上。如何并行化计算、分发数据和处理故障等问题使得原本简单的计算变得脢涩难懂，需要大量复杂的代码来处理这些问题。作为对这种复杂性的反应，他们设计了一个新的抽象，它允许我们表达我们试图执行的简单计算，但隐藏了库中并行化、容错、数据分布和负载平衡的混乱细节。这项工作的主要贡献是一个简单而强大的接口，它可以实现大规模计算的自动并行化和分布，结合这个接口的实现，可以在大型集群的商用pc上实现高性能。

第二篇是有关谷歌文件系统的。他们设计并实现了谷歌文件系统，这是一个可扩展的分布式文件系统，适用于大型分布式数据密集型应用程序。它在廉价的商用硬件上运行时提供容错功能，并为大量的客户端提供较高的聚合性能。虽然与以前的分布式文件系统有许多相同的目标，但他们的设计是由对应用程序工作负载和技术环境(当前和预期的）的观察驱动的，这些观察反映了与以前一些文件系统假设的明显背离。在文中，他们介绍了旨在支持分布式应用程序的文件系统接口扩展，讨论了他们设计的许多方面，并报告了从微基准测试和真实世界使用两方面的测量结果。首先，组件故障是正常现象，而不是昇常现象。该文件系统由数百甚至数千台存储机器组成，这些存储机器由廉价的商品部件构建，并由相当数量的客户端机器访问。这些组件的数量和质量实际上预示了一些组件在任何时候都无法正常工作且无法从当前的故障中恢复过来。因此，持续监控、错误检测、容错和自动恢复复必须是不可或缺的系统。其次，按照传统标准，文件是巨大的。多gb文件很常见。每个文件通常包含许多应用程序对象，如web文档。当经常使用由数十亿个对象组成的许多tb组成的快速增长的数据集时，即使在文件系统能够支持的情况下，管理数十亿个大约kb大小的文件也是非常笨拙的。第三，大多数文件都是通过添加新数据而不是覆盖现有数据来改变的。随机writeswithin文件实际上是不存在的。各种数据都具有这些特征。有些可能构成大型存储库，供数据分析程序扫描。有些可能是运行应用程序不断生成的数据流。第四，共同设计应用程序和文件系统API增加了灵活性，从而使整个系统受益。例如，他们将GFS的一致性模型放宽为极大地简化了文件系统，而没有强加繁重的应用程序的负担。我们还介绍了一个原子追加操作，使多个客户端可以追加并发的文件之间没有额外的同步。

第三篇是有关一种高效且广泛应用的社交图生成器FastSGG的，也不知道有没有翻译对，三篇论文中我重点阅读了这篇。以我的理解是说，当今随着社交网络的普及，尤其是在大数据时代，大规模社交图对于各种社交网络分析任务的算法评估是必要的。一个高效且可配置的社交图生成器比以往任何时候都更加重要，因为我们很难在各种场景中获得数十亿规模的真实社交图。在文中，他们提出了一种高效且广泛应用的社交图生成器FastSGG。FastSGG根据用户定义的描述目标社交图特性的配置生成社交图，这是在各种应用程序中生成图的一种灵活方式。生成方法主要包括两个步骤:确定源顶点的出度和确定目标顶点构造边。为了加速图的生成过程，提出了一种度分布生成(d2g)模型。d2g模型是在给定概率密度函数或概率质量函数的情况下，生成不同程度分布图的通用模型。此外，FastSGG生成图形的速度至少比最先进的图形生成器快4倍。此外，FastSGG的峰值内存使用量还不到最先进方法的七分之一。

接着上面所说，为了评估社会网络分析(SNA)任务的各种算法的性能和可伸缩性，现实世界和合成图都是必要的。然而，获取真实的网络图有三个主要的挑战，特别是对于大的图。首先，为了实现底层的大型网络，通常需要花费大量的时间来分析和处理社交媒体系统(例如，匿名化以避免触发隐私法)。具体来说，在特定时间点爬行给定的社交媒体后，只实现了一个真实的网络;网络的大小和结构，以及节点的标签和属性，都直接由生成的网络决定。例如，假设研究人员倾向于在特定时间对一个只有10亿用户的特定社交媒体进行快照。然而，这是很难保证在最终的网络。第三，在实际爬行网络中，大型复杂网络的特性经常遭到破坏。例如，一些抽样方法可以很容易地得到真实的网络图，但这些方法可能会改变现实世界复杂网络中固有的属性。

接着说了一些现存的一些算法的局限性。比如使用分层聚类或重排序/分块矩阵技术的团体检测算法是针对同构图进行的，而一些团体检测算法则是针对具有多个关系方面和多个顶点标签的异构图进行的。此外，现实世界的社区可以分为重叠社区和非重叠社区，许多社交应用程序的图表大小都呈指数级增长。因此，开发一种广泛使用、可扩展、高效的社交图生成器，以合成具有现实世界社交图属性的图，如小世界、社区结构、幂律分布，就显得十分重要。现有的合成图生成器不能满足文中例1中给出的所有要求。为了在不同的领域和应用中生成图形，提出了一些模式驱动的方法。这些方法，如gMark，使用精心设计的模式来覆盖图形中常见的特征，例如顶点和边的标签。然而，这些方法中的大多数都不是为社交图设计的，因为它们缺乏对生成具有社区结构的图的支持。此外，它们也不适合生成大规模的图形。例如，在他们的实验中，gMark需要花费6个多小时来生成一个具有100万个顶点的图，该图遵循幂律分布。

最终作者们提出一种可扩展的SGG（FastSGG）。生成大规模社交图有三个主要的技术挑战。首先，生成模型和算法应该足够高效，能够生成万亿规模的图。其次，生成算法应具有较低的时间和空间复杂度。最后，图的生成和社区的生成应该很容易结合起来，以生成带有社区结构的社交图。而FastSGG能够在一台商品机器上高效地生成万亿规模的异构社交图。FastSGG的生成方法在一个矩阵上执行，包括两个主要步骤:确定源顶点的出度和选择多个目标顶点来构造边。为了加速生成过程，他们提出了一种新的生成模型，称为度分布生成(D2G)模型。D2G模型提供了使用节点度概率生成图的原生支持。它是一个通用模型，这意味着如果给出pdf或pmf，我们可以使用该模型生成遵循不同度分布的图。使用这个模型，确定源顶点的出度或目标顶点只需要O(1)时间。因此，D2G适用于生成大规模图形。此外，他们可以使用该模型生成带有社区的图形。为了满足流图分析的需要，文中还提出了一种基于FastSGG的流图生成方法。表一总结了FastSGG和其他最先进的生成器的特点。要注意的是，只有由LFR生成的图的入度分布服从幂律分布。

他们的主要贡献：将图的矩阵表示的主对角线上的一个块作为一个社区，并提出了一种使用用户定义的模式生成社交图的方法。此外，他们设计了一种以流方式生成图的方法。他们提出了一个通用的生成模型，称为D2G模型，它可以用来生成遵循不同用户定义分布的图。理论分析表明，D2G模型能够有效地生成大规模图形。他们对FastSGG的性能进行了综合实验评价。实验结果表明，FastSGG能够生成具有真实世界社交图特征的社交图。结果还表明，FastSGG在效率和可伸缩性方面显著优于目前最先进的图生成方法。

我阅读到此，大致知道了这个FastSGG是干什么的，同时相比其它图生成器它的优势与创新之处。论文的其余部分大致介绍了社交图生成的模式的正式定义、社交图的生成方法、D2G模型的提出、使用D2G模型生成社交图的复杂性的分析、实验评价结果的报告、有关工作的讨论和论文的总结。其中的内容晦涩难懂，我无法深入解析。

最后小结一下，说实话读这三篇论文的过程中脑袋有点炸裂，咬着牙读完的。有的时候读到一半就不知道文章前面讲的是啥了，还得回过头重新从头开始，反反复复两三遍才略知一二。上面的内容基本都是从论文原文摘取的，因为不知道如果换其它的描述语意还对不对，因此我只是重新总结一下而已，伴随了我的一些可有可无的废话。

吕文江

17301130