**机器学习领域大数据科技报告**

李渤涟 3018205377

丰博达 3018001662

费雅 3018216191

张鼎 3018216092

**摘要**

机器学习领域在近十年内的飞速发展有目共睹，广泛的应用前景与极高的理论价值吸引了大批研究者投身其中，从而产出了大量优质的论文。从大量论文中把握机器学习的研究趋势，并寻找当前的研究热点，成为了研究者们提高工作效率与质量的重要途径。为此我们针对机器学习领域著名的数个会议近十年的论文大数据进行了分析，涉及数量统计、引用关系网络可视化、相关性分析以及频繁项集挖掘等大数据分析方法。我们基于分析结果对近10年机器学习领域的发展趋势进行了总结并对未来的发展方向进行了合理的预测，以期帮助初涉机器学习的学生或关注前沿的研究者更快更准确地了解研究热点与知名学者，并知晓不同子领域之间关系与发展脉络。

目录

[1. 分析计划 4](#_Toc12291)

[2. 网站数据采集与清洗 4](#_Toc30194)

[2.1. 数据源调研 4](#_Toc26928)

[2.2. 网络爬虫 5](#_Toc29384)

[2.3. 数据统计 7](#_Toc21688)

[3. 大数据分析方法 8](#_Toc30896)

[3.1. 年度论文录用数量变化分析 8](#_Toc7590)

[3.2. 引用关系网络的DAG图 8](#_Toc29079)

[3.3. 论文间相关性分析 10](#_Toc3320)

[3.4. 研究领域的频繁项集与强规则 11](#_Toc5675)

[4. 可视化效果及其分析 12](#_Toc17857)

[4.1. 会议论文数量柱状图 12](#_Toc32223)

[4.2. 关键词与作者词云 12](#_Toc18319)

[4.3. 热榜 14](#_Toc81)

[4.3.1 2019年各会议热榜 14](#_Toc4296)

[4.3.2 CVPR2010-2020年热榜 15](#_Toc31392)

[4.4. 引用网络的DAG图 16](#_Toc18707)

[5. 结果论证 18](#_Toc24754)

[5.1. 近10年中，机器学习领域的研究热度持续增长 18](#_Toc23579)

[5.2. 计算机视觉领域的发展要好于自然语言处理 19](#_Toc386)

[5.3. 少数研究者的工作总是为大多数研究者所参考 19](#_Toc28440)

[5.4. 机器学习工程方面的研究超越了其理论研究方向 19](#_Toc24290)

[5.5. 机器学习的各个方向之间尚未形成大量稳定的合作关系，仍然充满机遇 20](#_Toc12302)

[5.6. 深度学习方法逐渐成为了机器学习领域的主导 20](#_Toc15454)

[5.7. 机器学习子领域之间具有很强相关性 20](#_Toc13220)

[5.8. 可以通过论文大数据寻找优秀研究生导师 21](#_Toc26143)

[6. 参考文献 22](#_Toc32503)

[7. 分工情况 22](#_Toc30720)

[8. 附录 22](#_Toc1251)

[8.1. 数据源API 22](#_Toc28212)

[8.2. 论文数据集 23](#_Toc23174)

[8.3. 作者引用关系网络 23](#_Toc21548)

[8.4. 相关性分析数据（以如下8篇论文为例） 23](#_Toc17691)

[8.5. CVPR年度热榜（2010~2020） 24](#_Toc19357)

1. **分析计划**

我们的详细分析计划为：首先调研机器学习论文大数据的现状与分析的可行性，然后从预印本网站、数据库、会议官网等可靠渠道获取数据，然后对这些数据进行统计与分析，最后将分析结果可视化并得出我们的结论。详细的分析步骤如图1所示。

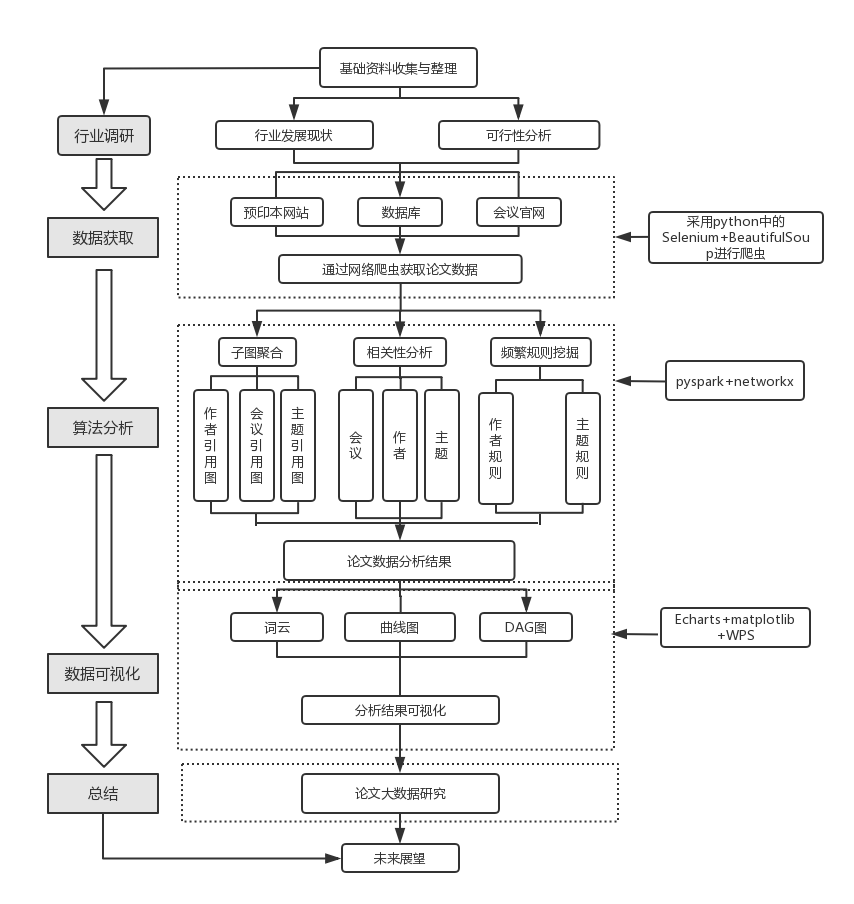
****

图1 技术路线

1. **网站数据采集与清洗**

**2.1. 数据源调研**

数据采集是进行大数据分析的前提也是必要条件，在整个流程中占据重要地位。数据采集的主要方法有：系统日志采集法、网络数据采集法以及其他数据采集法。在本次大作业中，我们采取网络数据采集法进行数据采集。

目前网络数据采集有两种方法一种是API，另一种是网络爬虫法。我们在此次大作业中的论文互相引用关系采用API方法进行采集，对于标题，作者，摘要等内容使用网络爬虫法。

API又叫应用程序接口，是网站的管理者为了使用者方面，编写的一种程序接口。该类接口可以屏蔽网站底层复杂算法仅仅通过简简单单调用即可实现对数据的请求功能。目前主流的社交媒体平台如新浪微博、百度贴吧以及Facebook等均提供API服务，可以在其官网开放平台上获取相关DEMO。我们此次查寻的网站dblp论文数据库也有相应的api，详见附录。

但是API技术毕竟受限于平台开发者，为了减小网站（平台）的负荷，一般平台均会对每天接口调用上限做限制，这给我们带来极大的不便利。为此我们在代码中调取python中的time库采用每爬取150条数据进行200s休止的做法。

**2.2. 网络爬虫**

网络爬虫（又被称为网页蜘蛛，网络机器人，在FOFA社区中间，更经常的称为网页追逐者），是一种按照一定的规则，自动地抓取万维网信息的程序或者脚本。另外一些不常使用的名字还有蚂蚁、自动索引、模拟程序或者蠕虫。（百度百科）最常见的爬虫便是我们经常使用的搜索引擎，如百度，360搜索等。此类爬虫统称为通用型爬虫，对于所有的网页进行无条件采集。通用型爬虫具体工作原理见图2-2。

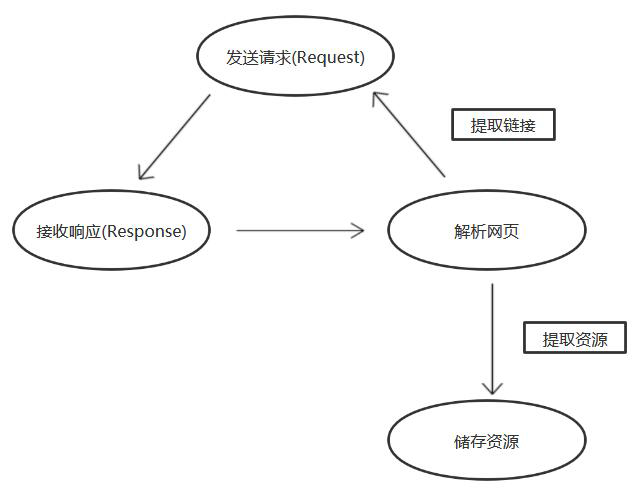


图2-2. 爬虫工作原理

给予爬虫初始URL，爬虫将网页中所需要提取的资源进行提取并保存，同时提取出网站中存在的其他网站链接，经过发送请求，接收网站响应以及再次解析页面，提取所需资源并保存，再将网页中所需资源进行提取…以此类推，实现过程并不复杂，但是在采集时尤其注意对IP地址，报头的伪造，以免被网管发现禁封IP（我就被封过），禁封IP也就意味着整个采集任务的失败。当然为了满足更多需求，多线程爬虫，主题爬虫也应运而生。多线程爬虫是通过多个线程，同时执行采集任务，一般而言几个线程，数据采集数据就会提升几倍。主题爬虫和通用型爬虫截然相反，通过一定的策略将于主题（采集任务）无关的网页信息过滤，仅仅留下需要的数据。此举可以大幅度减少无关数据导致的数据稀疏问题。

在具体实现上，我们采用python+selenium+BeatifulSoup进行爬虫。Selenium是一个自动化测试工具，可以驱动浏览器器执行特定的动作，如点击，下拉等。同时还可以获取浏览器当前呈现页面的源代码，可见即可爬。Beautiful Soup 提供一些简单的、python 式的函数用来处理导航、搜索、修改分析树等功能。它是一个工具箱，通过解析文档为用户提供需要抓取的数据。具体步骤如下：

①首先选取一部分精心挑选的种子URL；

例如：

<https://dblp.uni-trier.de/db/conf/cvpr/cvpr2020.html>

<https://dblp.uni-trier.de/db/conf/acl/acl2020.html>

②将这些URL放入待抓取URL队列；

③从待抓取URL队列中取出待抓取在URL，解析DNS，并且得到主机的ip ，并将URL对应的网页下载下来；

④使用Beautifulsoup以html格式解析并存储在存储进已下载网页库中，并将html页面中显示的数据如：作者，标题，摘要等进行爬取，并用特定的数据结构进行封装存储。

**2.3. 数据统计**

使用上述数据获取方法，我们先后获取了数个机器学习领域内的知名会议的数据，如表2-2-1所示。而在数据字段的选择上，我们以单篇论文为基本对象，获取了标题、作者、引用关系等，如表2-2-2所示。

|  |  |  |
| --- | --- | --- |
| 会议 | 年份范围 | 论文数量 |
| CVPR | 2010~2020 | 8214 |
| ACL | 2013~2019 | 1641 |
| IJCAI | 2017~2020 | 4692 |
| ICCV | 2011~2019 | 3015 |
| ICML | 2013~2020 | 4017 |
| NIPS | 2010~2020 | 6758 |

表2-2-1. 各会议数据统计年份范围

|  |  |
| --- | --- |
| 字段名称 | 格式说明 |
| 编号 | 数值 |
| 标题 | 字符串 |
| 摘要 | 字符串 |
| 作者 | 字符串集合 |
| 被引用关系 | 链接集合 |
| 引用关系 | 链接集合 |
| 关键词 | 字符串集合 |

表2-2-2. 数据字段与格式

1. **大数据分析方法**
   1. **年度论文录用数量变化分析**

这是我们分析的第一步，我们期望统计出各大会议论文的数量变化，以此观察领域发展的趋势，以及发展的速度等，由于数据量较大，所以我们采用了MapReduce算法。构建论文的统计，利用pyspark完成统计，同时也可以看子领域的发展情况，甚至看各大会议的重要性的变化。

MapReduce是一种编程模型，用于大规模数据集（大于1TB）的并行运算。概念"Map（映射）"和"Reduce（归约）"，是它们的主要思想，都是从函数式编程语言里借来的，还有从矢量编程语言里借来的特性。它极大地方便了编程人员在不会分布式并行编程的情况下，将自己的程序运行在分布式系统上。 当前的软件实现是指定一个Map（映射）函数，用来把一组键值对映射成一组新的键值对，指定并发的Reduce（归约）函数，用来保证所有映射的键值对中的每一个共享相同的键组。

* 1. **引用关系网络的DAG图**

在绘制引用关系网络的DAG图（有向无环图）的时候，我们考虑到每篇论文都有引用的文章和被引用的文章两个集合，但实际上两者都是在一个闭合的引用关系网络里，所以两者是等价的，若顺着一个集合去寻找，即可找全所有关系，因此，我们选用了被引用这一个集合。

之后，我们希望获取作者之间的引用关系，期望发掘作者之间的朋友，指导等相关讯息，考虑到数据集非常庞大，我们选择借用pyspark的MapReduce算法，采用聚类、Wordcount等手段，将其汇总，进行可视化，进行进一步的讨论。在绘制引用关系的图时候，我们也在思考每篇论文的一作二作三作甚至更多作者之间的关系，最后，我们选择直接使用在各大网站上，给予一作二作三作的权值来作为引用的边权，最后通过边权来看引用关系的强弱，以及其他一些有趣的消息。

我们也去寻找了关键词之间引用关系，期望挖掘出他们之间的内在联系，甚至观察其前后继承关系，进一步推测其延申的可能性。在分析的时候，我们也借用了pyspark的MapRudece算法，按年份统计了其引用关系变化， 期望发现更多的信息，但每篇文章的关键词也有很多，我们也要给予不同的权值，最后，我们选择对摘要进行关键词抽取，从摘要中，借用文本标签算法，我们给不同的关键词赋予不同的权值，再绘制图像，为后面可视化的分析提供便利。

最后，我们也想去看看不同会议间的引用关系，看看会议的级别、会议的类别等因素之间的影响。我们同样采用了上述的方法，但在这一步，我们也想看看是否一定是低等级的会议引用更多高等级的会议，所以这里并没有很细致去给边权赋值，而是直接赋值成1，直接统计数量，以此来观察我们是否可以发现更多的有趣的知识。

在本处使用的算法，我们直接使用pyspark封装好的MapReduce，之后的聚合由自己去设计，权值的赋予则是由我们讨论完成的，设计的自然语言处理相关的算法，我们选择调用百度AI的接口，使用其封装好的算法简化我们的工作，让我们把更多的精力放到分析上。

* 1. **论文间相关性分析**

在讨论论文相关性的时候，我们思考了很多方面，但最后还是选择作者，关键词，会议，去挖掘它们之间的关系，由于它们都是文字，不在一个度量空间内，所以我们也在思考如何转换，最后选择了使用pyspark的word2vec算法，借用自然语言处理，将其转换成多维数值，考虑到算力的局限性和性能的要求，我们最后选择将每个论文拼成长度为24的向量，为下一步调用相关性算法提供遍历。

之后我们选择使用pyspark.ml.Correlation包，调用其相关算法，计算皮尔逊指数矩阵，以此观察这些论文的相关性，但本身由于论文数量过于庞大，所以这个矩阵不可能生成所有论文的相关性分析，需要进行一些挑选，生成我们所关注的一些论文的相关性矩阵，这样更适合我们去分析我们所关注一些领域的问题。

word2vec是一个自然语言处理的算法，用处是词语转换成向量。原理如下：

大部分的有监督机器学习模型，都可以归结为输入数据到预测结果的映射f: X → Y。在自然语言处理中，把x看做一个句子里的一个词语，y 是这个词语的上下文词语，那么这里的 f，便是 NLP 中经常出现的“语言模型”（language model），这个模型的目的，就是判断 (x,y) 这个样本，是否符合自然语言的法则，更通俗点说就是：词语x和词语y放在一起，是不是人话。

Word2vec 正是来源于这个思想，但它的最终目的，不是要把 f 训练得多么完美，而是只关心模型训练完后的副产物——模型参数（这里特指神经网络的权重），并将这些参数，作为输入 x 的某种向量化的表示，这个向量便叫做词向量。

皮尔逊相关系数的变化范围为-1到1。 系数的值为1意味着X和Y可以很好的由直线方程来描述，所有的数据点都很好的落在一条[直线](https://baike.baidu.com/item/%E7%9B%B4%E7%BA%BF/4876" \t "https://baike.baidu.com/item/%E7%9A%AE%E5%B0%94%E9%80%8A%E7%9B%B8%E5%85%B3%E7%B3%BB%E6%95%B0/_blank)上，且Y随X增加而增加。系数的值为−1意味着所有的数据点都落在直线上，且Y随着X的增加而减少。系数的值为0意味着两个变量之间没有线性关系。

更一般的, 我们发现，当且仅当Xi和Yi均落在他们各自的均值的同一侧， 则如果Xi和Yi同时趋向于大于，或同时趋向于小于他们各自的均值，则相关系数为正。如果Xi和Yi趋向于落在他们均值的相反一侧，则相关系数为负。

我们的相关性分析结果详见附录。

* 1. **研究领域的频繁项集与强规则**

在挖掘领域的频繁项集与强规则时，我们曾想过使用Apriori算法，但因其速度太慢，最后还是选择使用pyspark的FPGrowth算法，期望可以挖掘出研究领域的发展关系，期望可以发现更多可以相互关联的可能性。

Apriori算法是第一个关联规则挖掘算法，也是最经典的算法。它利用逐层搜索的迭代方法找出数据库中项集的关系，以形成规则，其过程由连接（类矩阵运算）与剪枝（去掉那些没必要的中间结果）组成。该算法中项集的概念即为项的集合。包含K个项的集合为k项集。项集出现的频率是包含项集的事务数，称为项集的频率。如果某项集满足最小支持度，则称它为[频繁项集](https://baike.baidu.com/item/%E9%A2%91%E7%B9%81%E9%A1%B9%E9%9B%86/1573014" \t "https://baike.baidu.com/item/APRIORI/_blank)。

FPGrowth是韩家炜教授在2000年提出的频繁项集挖掘算法，其中FPTree是使得这一算法相比Aprioris等算法较为高效的关键数据结构，FPTree将数据库中的所有事务(Transactions)高度压缩成树的路径，所有的频繁项(Frequent Items, FI)都成为树的一个节点，每个节点都拥有相应的计数，代表该FI在数据库中出现的次数，其中叶子节点的计数等于前向遍历路径中的FI出现在数据库中的次数。因此所有的挖掘工作都以最初的FPTree为中心展开，而在构建一棵FPTree时，核心步骤在于对每一条事务进行降序排序。

1. **可视化效果及其分析**
   1. **会议论文数量柱状图**

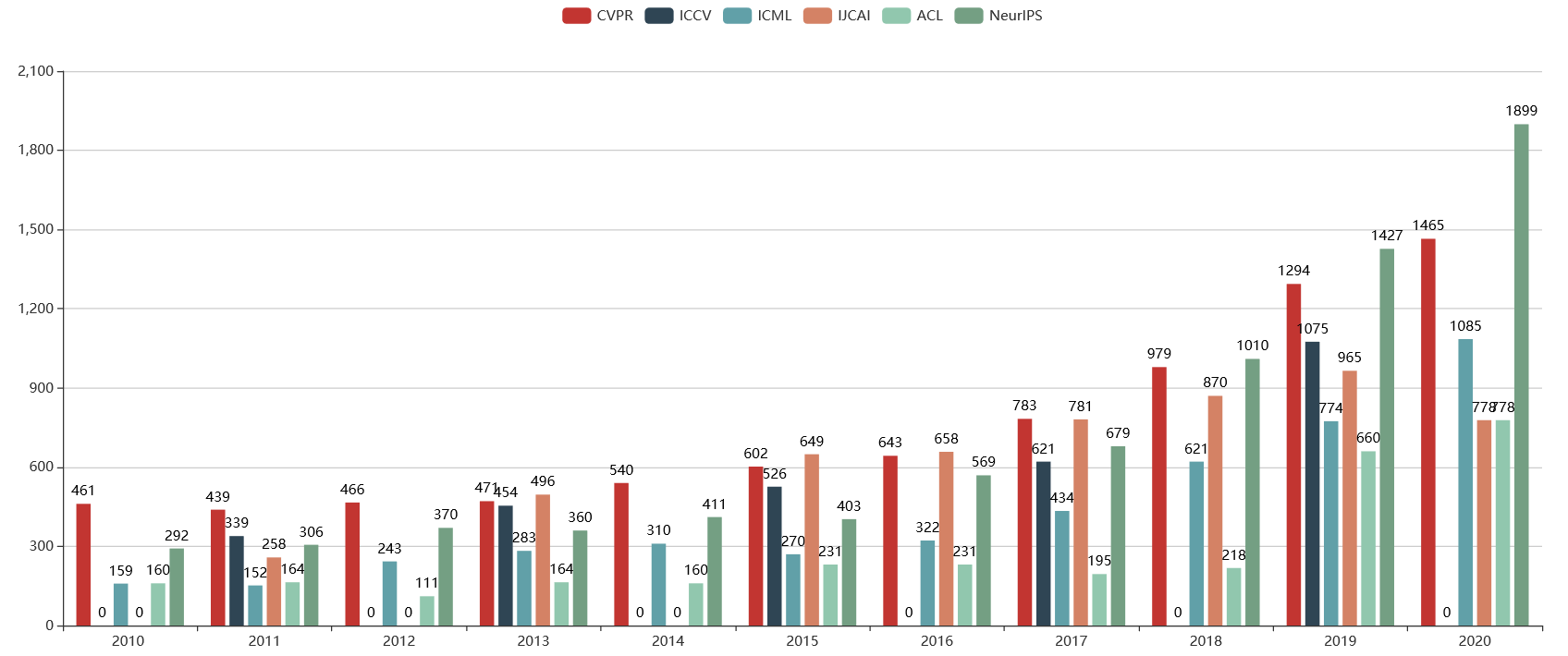
****

图4-1 各会议年度录用论文数量变化

可以看到，CVPR的录用论文数量逐年上升，而NIPS在2019超过了CVPR，并且上升趋势也远高于CVPR。同时，ACL的录用论文数量一直低于CVPR，未能有超越的迹象，除了IJCAI，其他的论文录取量都是呈现上升的趋势，一直在增多，而IJCAI却有些起伏，在2020年录用论文数量甚至出现的负增长。

* 1. **关键词与作者词云**

为了查看5年来机器学习研究领域的热点的变化，我们接下来将重点展示2015年与2019年的作者词云与关键词词云。

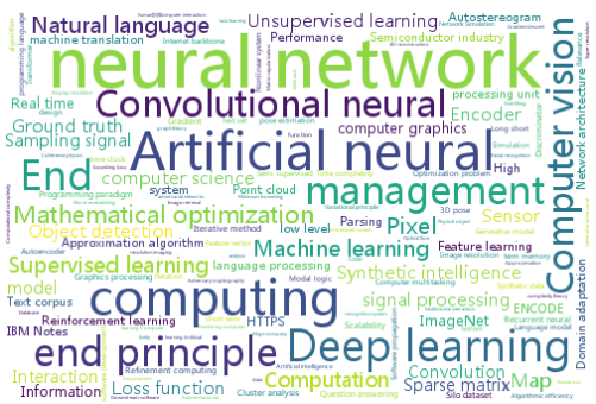
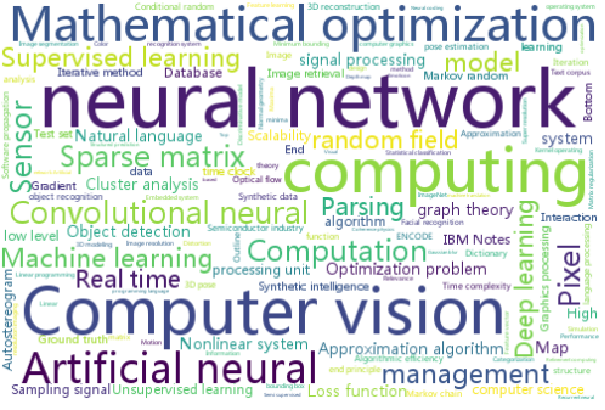


图4-2-1. 2015年（左）与2019年（右）论文关键词词云

在关键词词云中，我们发现15年和19年的存在感高的关键词并无太大变化，说明机器学习领域的计算机视觉、神经网络等主题仍是学者主要的研究方向。同时占比较少的关键词有较多更新，说明5年来有一些新的研究方向出现。



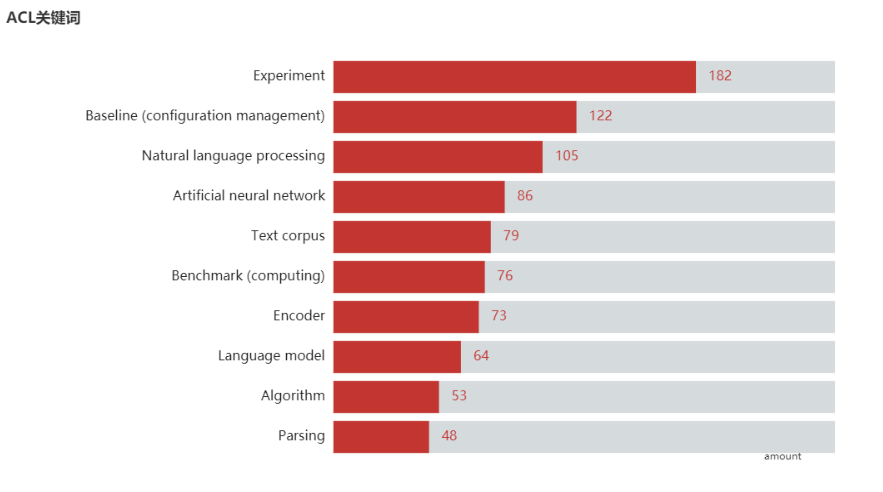
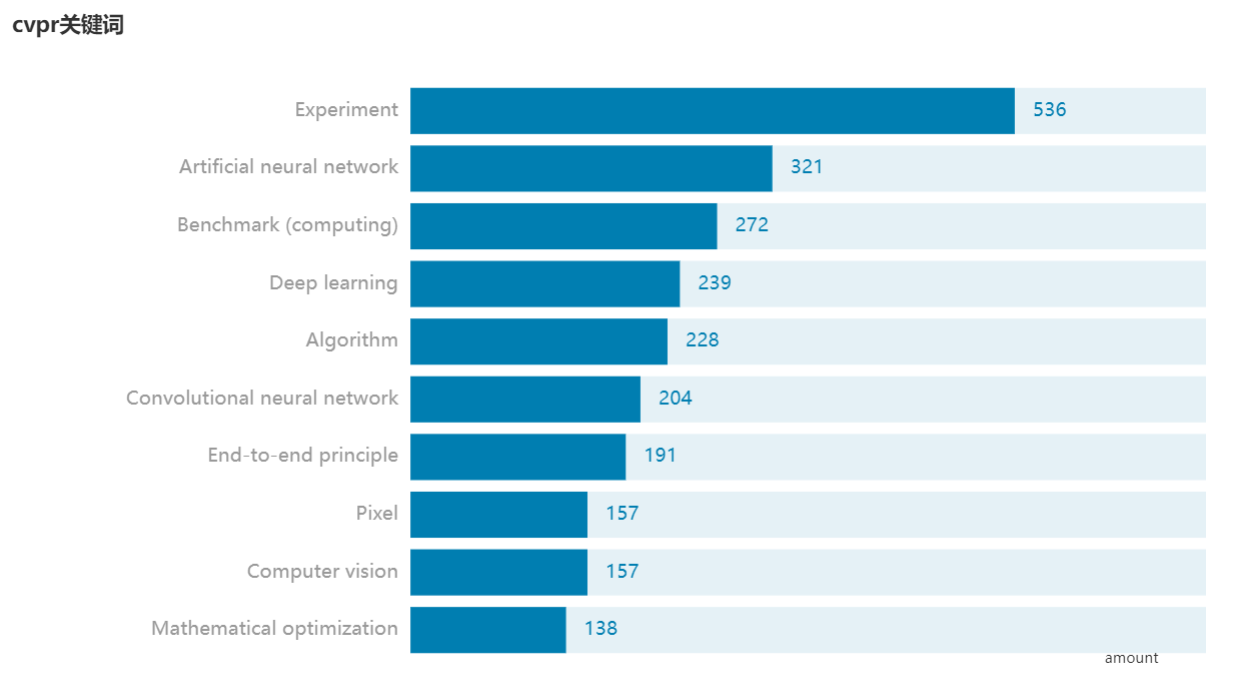
图4-2-2. 2015年（左）与2019年（右）论文作者词云

在作者词云中，我们发现一位名叫“X. Wang”的作者非常突出，在2019年的CVPR上发表了27篇文章，出于好奇，我们在Google Scholar上对该作者进行了检索，结果发现原来X. Wang是多位作者的名称缩写，这27篇文章是不同的作者发表的。但是在ACL会议中，一位名叫“Z. Liu”的作者也在霸榜，依照此思路，我们也一开始认为此人为多个作者，但实际上经过统计，发现该作者为清华大学刘知远老师，在ACL2019中总共发表了12篇长文，一骑绝尘，令人惊叹。

由此可见，我们这个方法统计作者的时候，存在一些瑕疵，从根本上看，这个问题与自然语言处理中的实体识别问题很相似，都是由于同样的文字却指代了多个实体（在本次作业中是指代了多个作者），如何解决这一问题，值得我们深思，我们想到的方法是使用authorId替换作者，但是在爬取的数据中部分作者的authorId为null，因此我们仍然使用作者姓名进行分析。

* 1. **热榜**

**4.3.1 2019年各会议热榜**

****

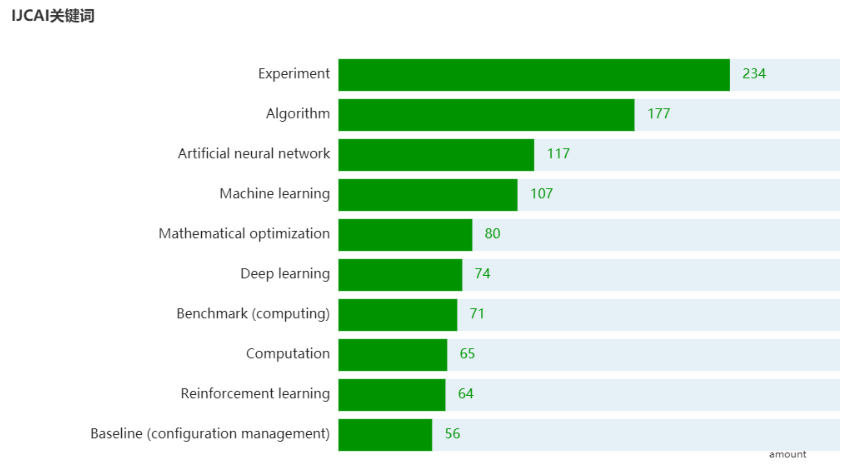
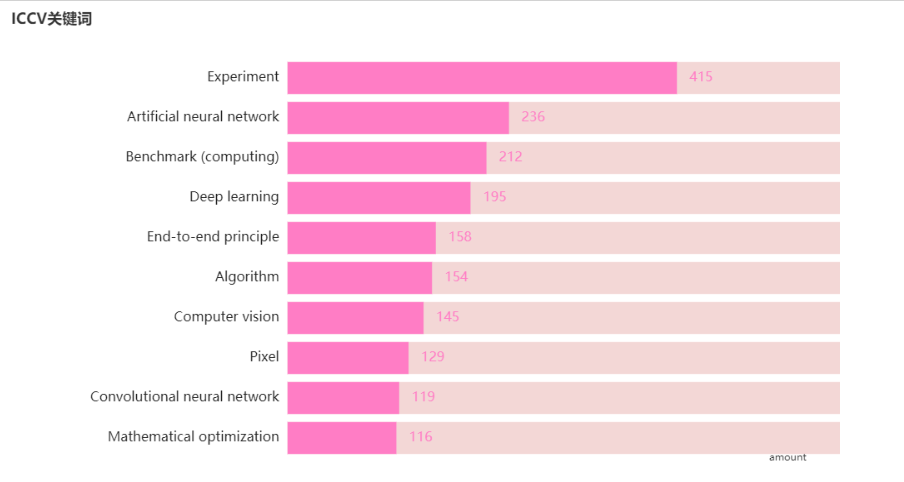
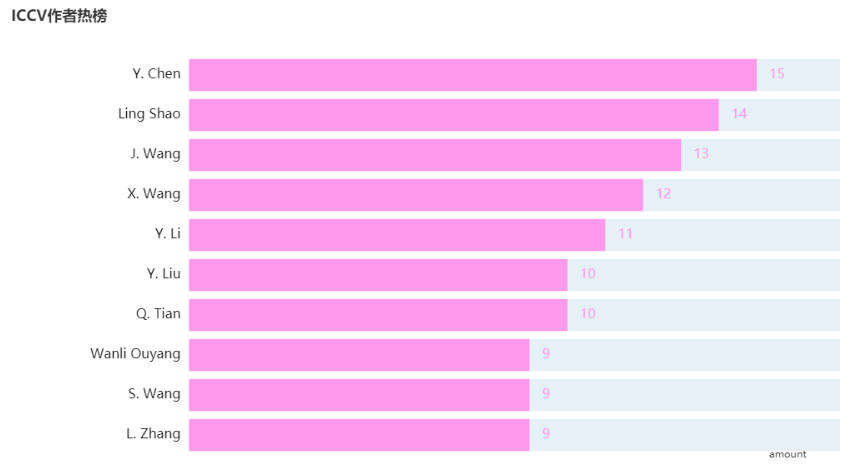
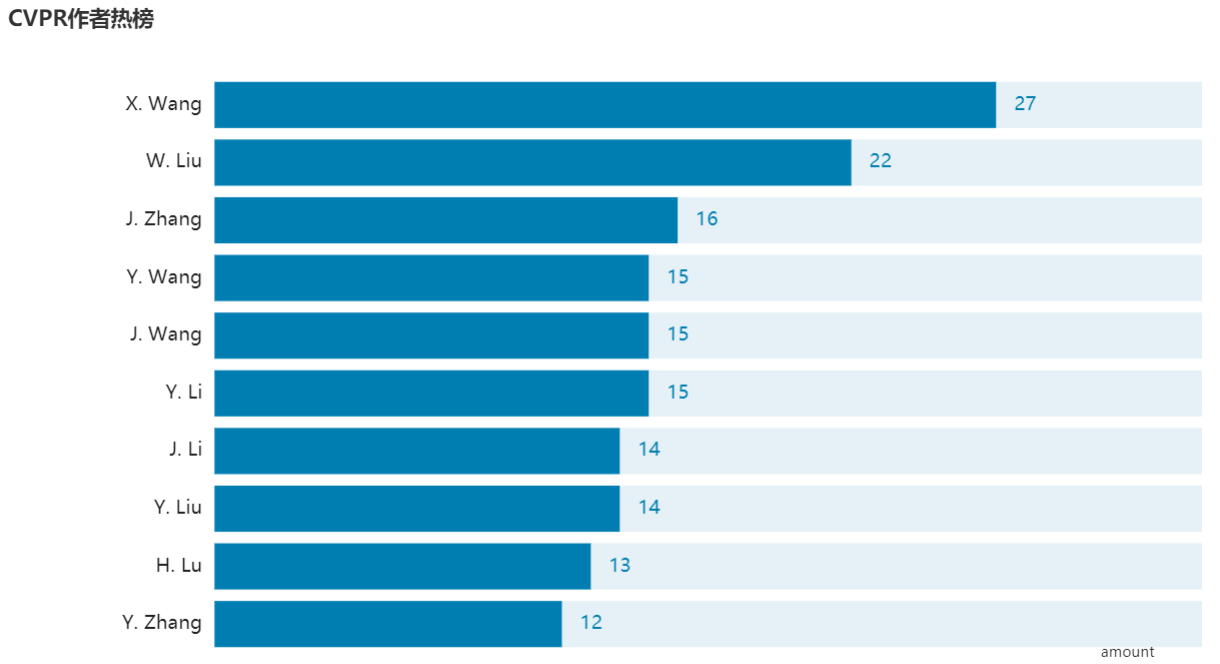
****

图4-3-1-1. 各会议关键词热榜

通过对2019年各大会议的Top10关键词的对比，我们可以看出各大会议核心关注点都保持着一致，只是有一些特别的偏向性，由此即可给不同作者提供发论文的更好方向，有更大几率被录用。

****

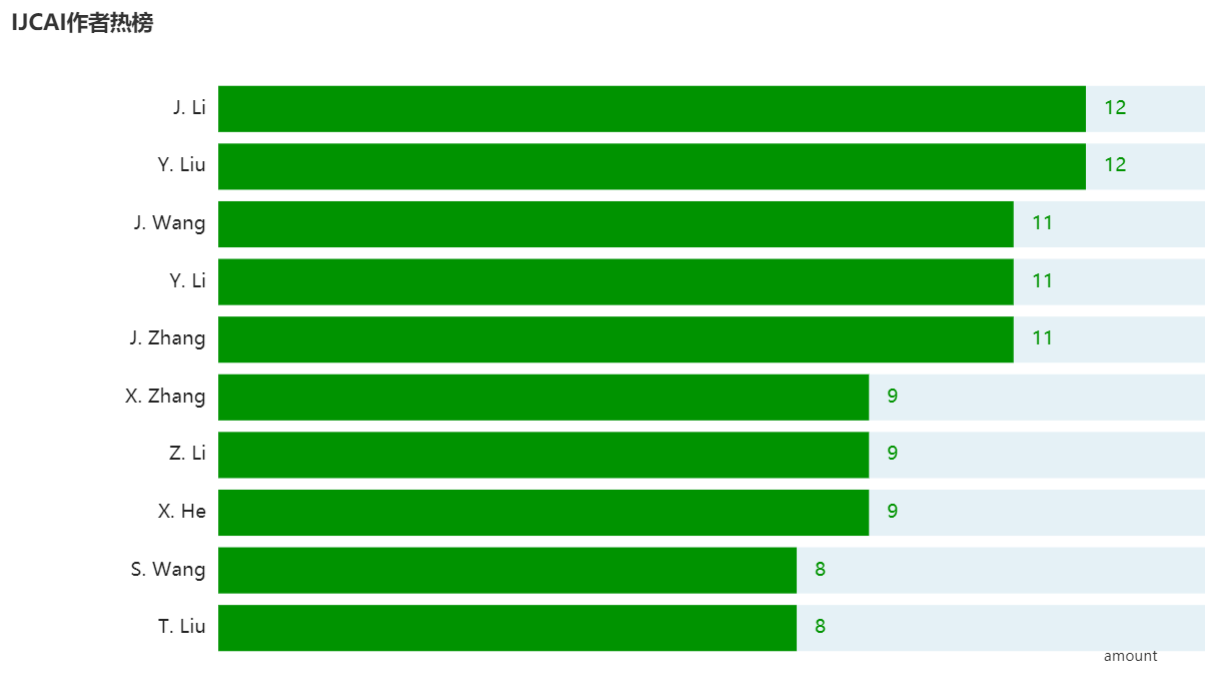
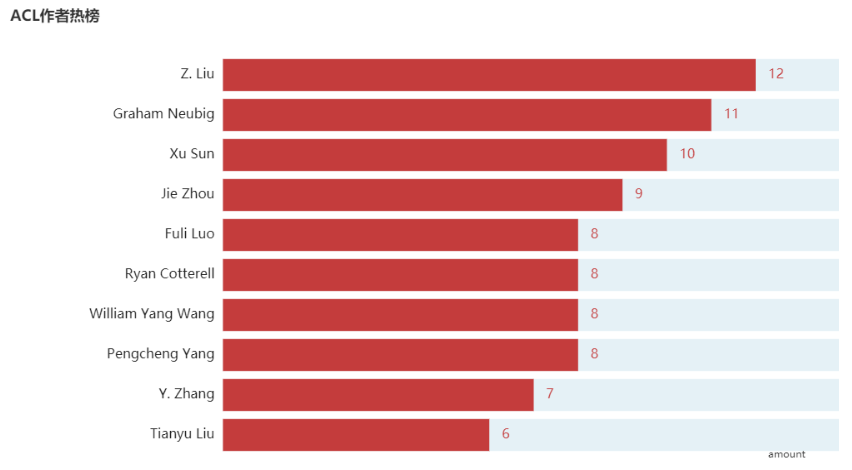
****

图4-3-1-2. 各会议作者热榜

通过对比2019年各个会议作者发文数量热榜，我们发现不同研究人员选择发表论文的会议有着很大的不同，这可能是因为不同会议的主流研究方向有着很大的区别，因此从一位研究者的投稿倾向上就可以大致了解其研究兴趣。

**4.3.2 CVPR2010-2020年热榜**

我们选取收录论文丰富、投稿作者多的CVPR会议分析其2010-2020年的关键词和作者热榜。可视化效果为动态条形图Top10排行榜，图4-3-2-1和图4-3-2-2为2010年截图，动态效果见可视化代码中的cvpr-topic.html和cvpr-author.html。

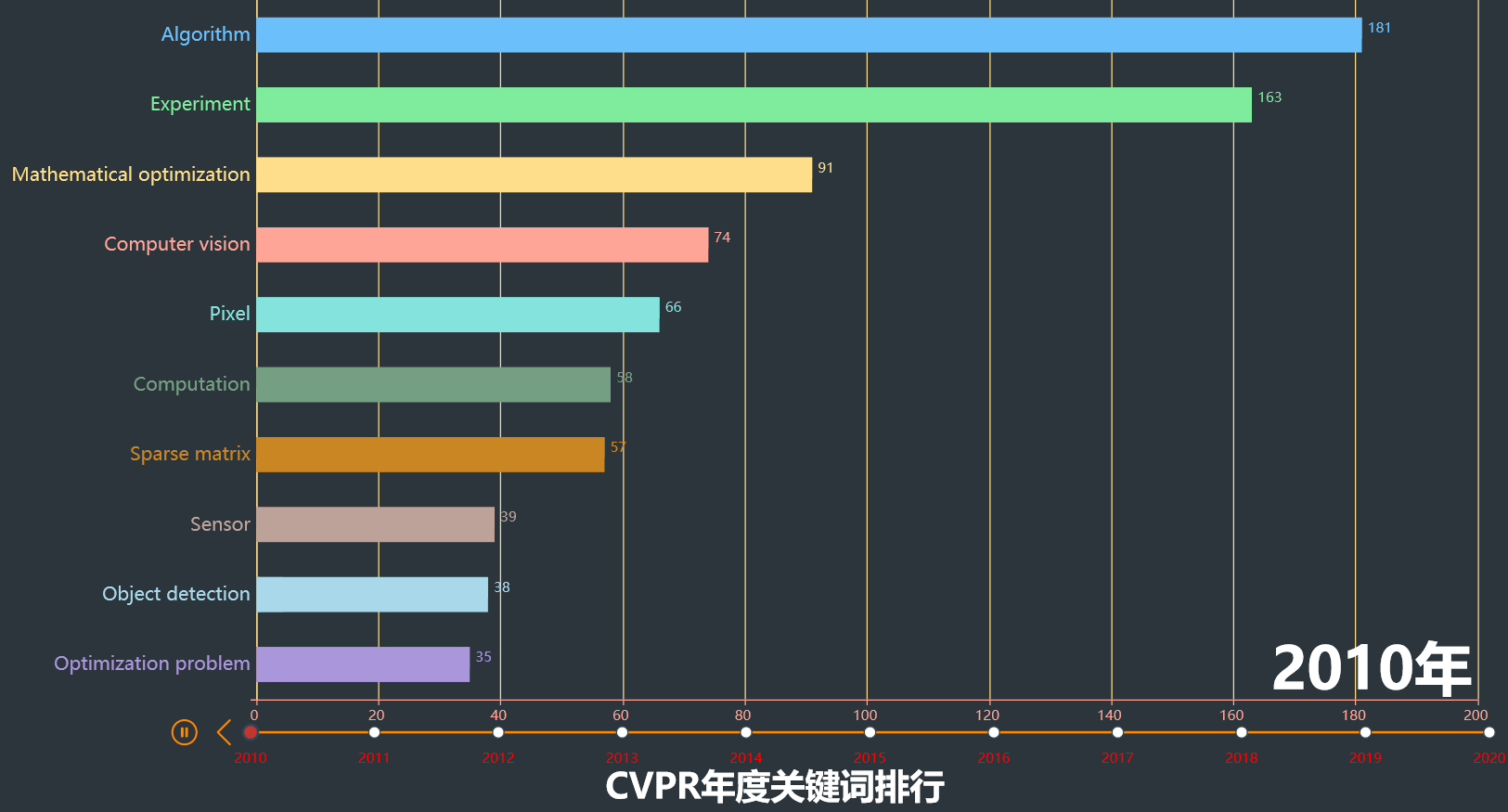
****

图4-3-2-1. CVPR2010-2020年关键词热榜（其动态原图详见附录）

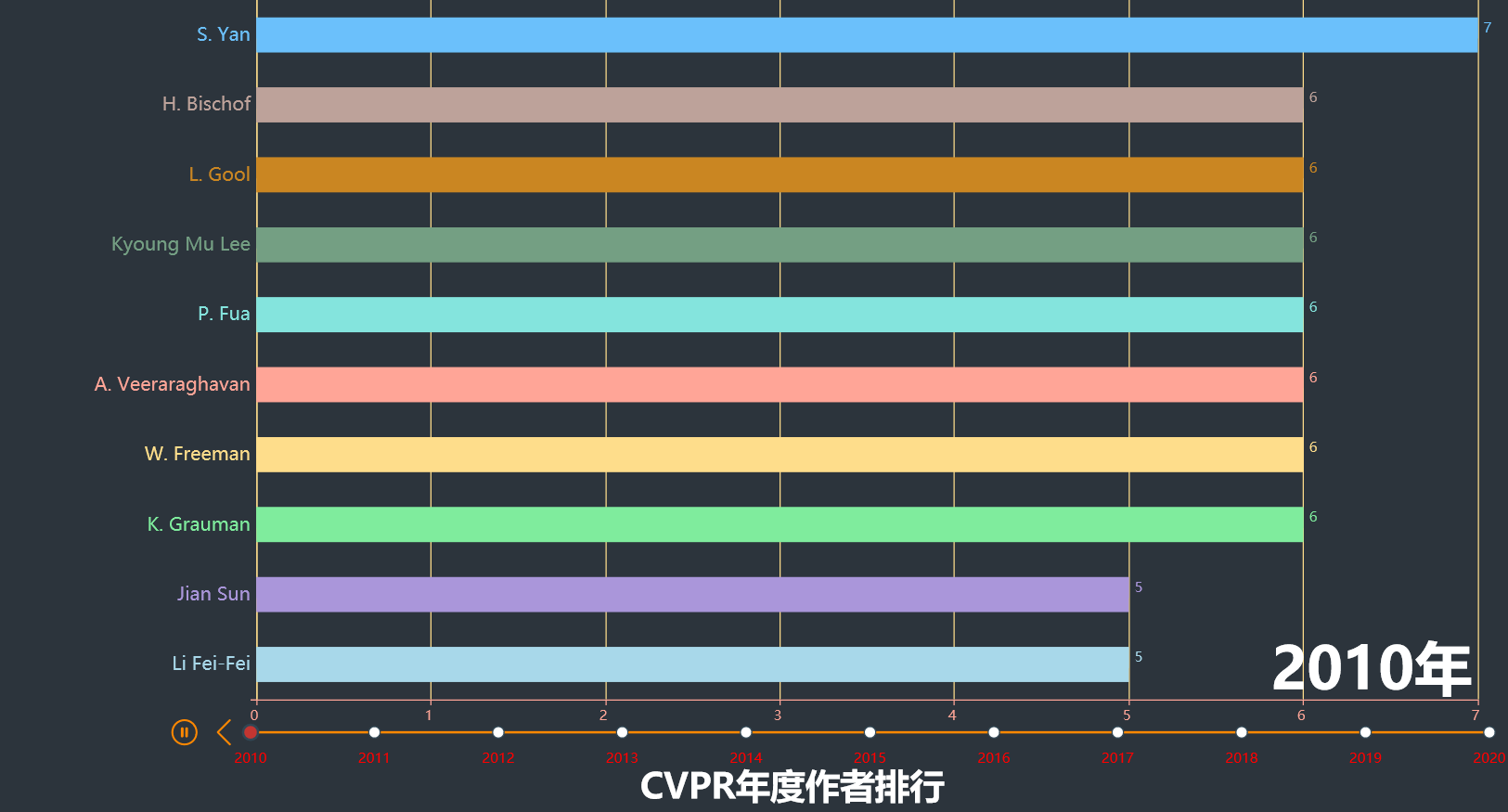
****

图4-3-2-2. CVPR2010-2020年作者热榜（其动态原图详见附录）

我们发现CVPR各年会议的关键词热榜变化不大，但Top10中“Algorithm”原本位于2010-2017年的Top3，在2018年后下滑至排名中游；“Mathematical optimization”原本位于2010-2015年的第3、4位，在2016年后下滑至排名末位，说明在机器学习领域研究者对于纯数学理论的兴趣呈现出逐年下滑的趋势；“Artificial neural network”和“Benchmark”从不入榜到排名前列，“Convolutional neural network”在2016-2018年迎来短暂高峰，说明在机器学习领域研究者越来越重视对工程性技术的研究。

作者各年的Top10变化很大，每年会有新的作者出现，说明在机器学习领域每年都有新的科研人才涌现。“X. WANG”的在近几年的霸榜现象在作者词云中有解释。

* 1. **引用网络的DAG图**

我们基于引用数据分别得到了作者、关键词，以及会议之间的应用关系网络。其中作者之间的引用关系网络详见附录，关键词与会议的应用网络如下所示。

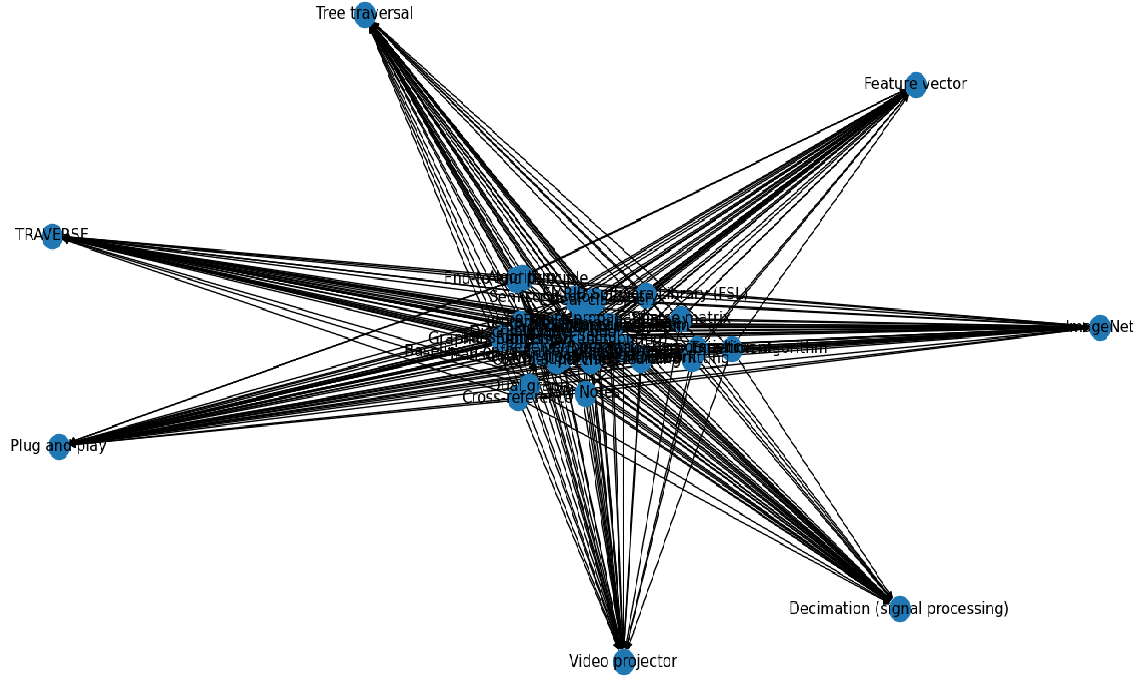


图4-4-1. 关键词引用关系网络

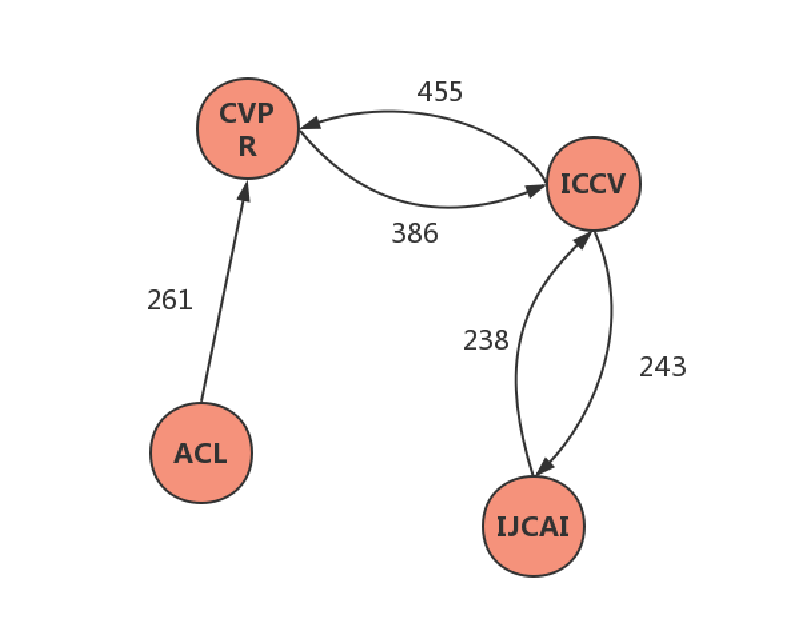


图4-4-2. 会议引用关系网络

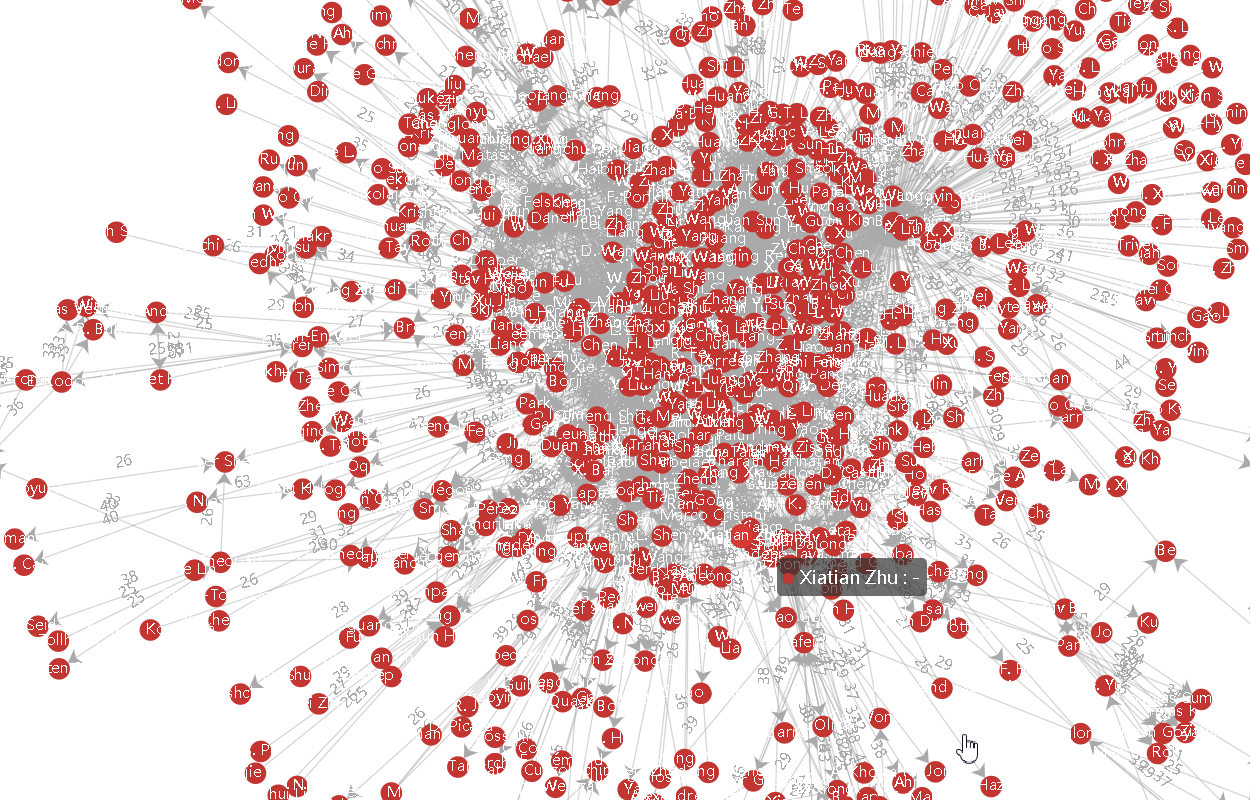


图4-4-3. 作者引用关系网络（其动态原图详见附录）

在分析了关键词、会议和作者的引用关系后，我们可以看到各大会议之间的密切联系,CVPR确实是计算机视觉的第一会议，其被引用量也是令人瞩目，ACL作为自然语言处理的会议经常引用CVPR，这也是一个令人惊讶的事实。关键词之间的相互引用也可以看出各个研究领域的发展过程，可以看到关键词之间的借鉴关系，或许也可以为研究者提供新的研究思路，为社会带来更多的帮助。而作者之间的引用关系也呈现出明显的集群效应，少数作者成为了整个网络结构的核心，是整个学界的主要思想来源。

1. **结果论证**

基于上述的分析，我们发现机器学习论文呈现出以下几种趋势。

* 1. **近10年中，机器学习领域的研究热度持续增长**

我们经过分析，发现机器学习领域各大会议录用的数量持续爆发式增长，CVPR的录用数量从2010年的400篇，到2020年的1200篇，虽然ijcai的数量有下降趋势，但2020的数量还是高于2010年，足以看出机远器学习的研究热度不仅在持续增长，并有一种爆发的趋势。因此，我们甚至可以推测，未来几年，机器学习领域的研究热度仍会上升。

* 1. **计算机视觉领域的发展要好于自然语言处理**

我们发现，在每年的各大会议录用数量的横向对比中,无论是CVPR，还是ICCV，亦或是有衰落现象的IJCAI，都明显多于ACL，同时，在各大会议的引用关系图中，也能看到ACL引用的计算机视觉的技术更多，说明自然语言处理对计算机视觉的依赖。

这一现象说明，在近10年里，计算机视觉的成果要远多于自然语言处理的成果，因此，我们可以推断计算机视觉领域的发展要好于自然语言处理。

* 1. **少数研究者的工作总是为大多数研究者所参考**

我们在分析引用图的时候，可以明显看到，整个网络呈现分中心聚集的现象，在庞大的网络中，有几个中心点，这体现出，这几人的工作给大多数人都提供了便利，因此我们可以得出结论：少数研究者的工作总是为大多数研究者所参考。

* 1. **机器学习工程方面的研究超越了其理论研究方向**

我们还发现在工程领域的机器学习研究的火热程度超过了理论研究，以CVPR为例，2010年关键词“数值优化”的出现频率位列热榜第3，而到2018年，“数值优化”则跌至热榜第10位。

这一趋势说明，机器学习中纯数学方法的研究在近10年内得到的关注越来越少，而机器学习算法的实际应用与产业化落地正在拥有越来越大的价值，因此我们预测在未来的一段时间内，应用机器学习仍将主导机器学习研究。

* 1. **机器学习的各个方向之间尚未形成大量稳定的合作关系，仍然充满机遇**

我们发现作者之间的引用关系与研究防线之间的引用关系还未形成大量稳定的关系，如下表所示，作者之间的单次引用数量最多，而且占到了73.5%的比例。

|  |  |  |
| --- | --- | --- |
| 引用次数 | 频数 | 占比 |
| 1 | 2800390 | 73.5% |
| 2 | 581051 | 15.3% |
| 3 | 192401 | 5.1% |
| 4 | 88753 | 2.3% |
| 5 | 46647 | 1.2% |

表5-5. 引用次数统计表

* 1. **深度学习方法逐渐成为了机器学习领域的主导**

我们发现在年度关键词的热榜中，有关深度学习的词汇的排名逐年上涨。“人工神经网络”、“卷积神经网络”、“深度学习”等关键词在2015年前后首次进入前10榜单，并在2017年就达到了第2名。

* 1. **机器学习子领域之间具有很强相关性**

我们基于FPGrowth算法得到了论文关键词的频繁2项集与强规则，如表5-7-1与表5-7-2所示。

|  |  |
| --- | --- |
| 频繁2项集 | 频度 |
| Computer vision，Artificial neural network | 381 |
| Deep learning，Artificial neural network | 805 |
| Convolutional neural network，Deep learning | 294 |
| Convolutional neural network，Computer vision | 297 |
| Convolutional neural network，Artificial neural network | 961 |
| End-to-end principle，Artificial neural network | 367 |
| Optimization problem，Mathematical optimization | 403 |

表5-7-1. 关键词的频繁2项集

通过上表我们可以知道，上述频繁2项集代表两个机器学习子领域经常在同一篇文章中出现，说明机器学习子领域之间具有很强相关性。

|  |  |  |
| --- | --- | --- |
| 强关联规则 | 置信度 | 提升度 |
| Convolutional neural network => Deep learning | 0.203039 | 1.906544 |
| Convolutional neural network => Computer vision | 0.205110 | 1.743439 |
| Convolutional neural network => Artificial neural network | 0.663674 | 4.046917 |
| Mathematical optimization => Optimization problem | 0.233083 | 5.877453 |
| End-to-end principle => Artificial neural network | 0.315563 | 1.924225 |
| Artificial neural network => Deep learning | 0.342116 | 3.212491 |
| Artificial neural network => Convolutional neural network | 0.408415 | 4.046917 |
| Computer vision => Artificial neural network | 0.225711 | 1.376328 |
| Deep learning => Artificial neural network | 0.526832 | 3.212491 |

表5-7-2. 关键词的强规则

通过上表我们可以知道，当强规则左边的关键词出现在文章中，强规则右边的关键词也很有可能出现在该文章中，所以当我们查看文章的一些关键词后可以合理推测另一些关键词也会出现在文章中，可以帮助我们更高效地找到想要的文档。

**5.8. 可以通过论文大数据寻找优秀研究生导师**

对于刚入学科研能力强的究生，如果想要从事科研工作，选一个科研能力强的导师是很重要的。我们此次分析，可以为此方面提出一些建议。首先，我们可以统计想要从事领域的顶级会议或者期刊中，本校教师发表论文数据数目，从中选出发表论文数比较多的课题组和教师。其次，我们可以在谷歌学术上查找并比较自己心仪导师们的h-index指数。最后，可以在DBLP，arxiv，网站上寻找自己心仪的导师最新发表的论文进行精读，了解该老师的研究进展。

综上所述，我们的工作可以帮助研究生们在对导师不了解的情况下，对导师的科研水平有初步的了解，进而选择科研能力较强的老师，开启更美好的科研生涯。

1. **参考文献**
2. <https://www.cnblogs.com/tangweijqxx/p/10803864.html>
3. <https://blog.csdn.net/mmc2015/article/details/50988375?locationNum=4&fps=1>
4. Mikolov T , Chen K , Corrado G , et al. Efficient Estimation of Word Representations in Vector Space[J]. Computer Science, 2013.
5. Han J , Pei J . Mining frequent patterns by pattern-growth: methodology and implications[J]. Acm Sigkdd Explorations Newsletter, 2000, 2(2):14-20.
6. Dean J , Ghemawat S . MapReduce: simplified data processing on large clusters[J]. Communications of the ACM, 2008, 51(1):p.107-113.
7. **分工情况**

|  |  |  |  |
| --- | --- | --- | --- |
| 姓名 | 学号 | 主要负责的工作 | 备注 |
| 李渤涟 | 3018205377 | 统筹进度、安排任务、Spark环境配置、代码编写与运行、课上展示 | 组长 |
| 丰博达 | 3018001662 | 代码编写与运行、算法原理推导、报告书写 |  |
| 费雅 | 3018216191 | 数据获取、数据可视化、报告的后期完善 |  |
| 张鼎 | 3018216092 | 数据获取、数据可视化、工作场地管理 |  |

表7. 分工情况

1. **附录**
   1. **数据源API**

[https://api.semanticscholar.org/v1/paper/10.24963/IJCAI.2020/1?include\_unknown\_references=true&mailto=ajax@dblp.org](https://api.semanticscholar.org/v1/paper/10.24963/IJCAI.2020/1?include_unknown_references=true&mailto=ajax@dblp.org。)

* 1. **论文数据集**

我们获取的全部论文数据都在随报告附上的压缩包dataset.7z中，此外也可以访问<https://github.com/lblaoke/PaperBigData-for-MachineLearning>以获取我们的数据。（由于我们的数据量很大，推荐直接访问上述链接）

* 1. **作者引用关系网络**

作者之间的引用关系网络详见可视化代码中的author-all.html和author-two.html。

* 1. **相关性分析数据（以如下8篇论文为例）**

|  |  |
| --- | --- |
| 编号 | 标题 |
| 1 | Object-graphs for context-aware category discovery |
| 2 | Grouplet: A structured image representation for recognizing human and object interactions |
| 3 | Modeling mutual context of object and human pose in human-object interaction activities |
| 4 | The chains model for detecting parts by their context |
| 5 | Detecting and sketching the common |
| 6 | High performance object detection by collaborative learning of Joint Ranking of Granules features |
| 7 | P-N learning: Bootstrapping binary classifiers by structural constraints |
| 8 | 3D Scene priors for road detection |

表8-4-1. 论文题目

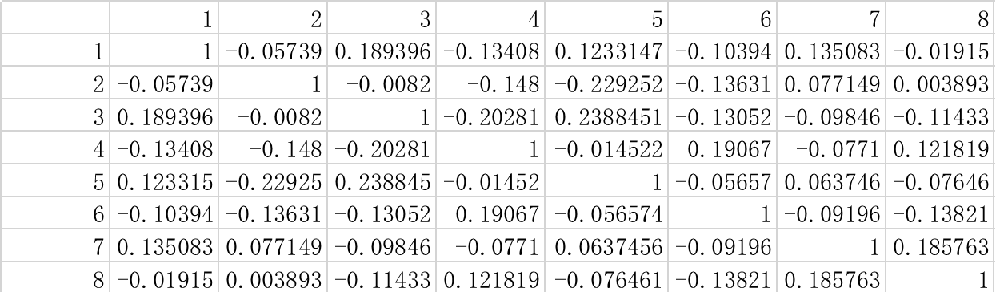


表8-4-2. 相关性矩阵

* 1. **CVPR年度热榜（2010~2020）**

CVPR年度作者热榜与年度关键词热榜详见可视化代码中的cvpr-author.html与cvpr-topic.html。