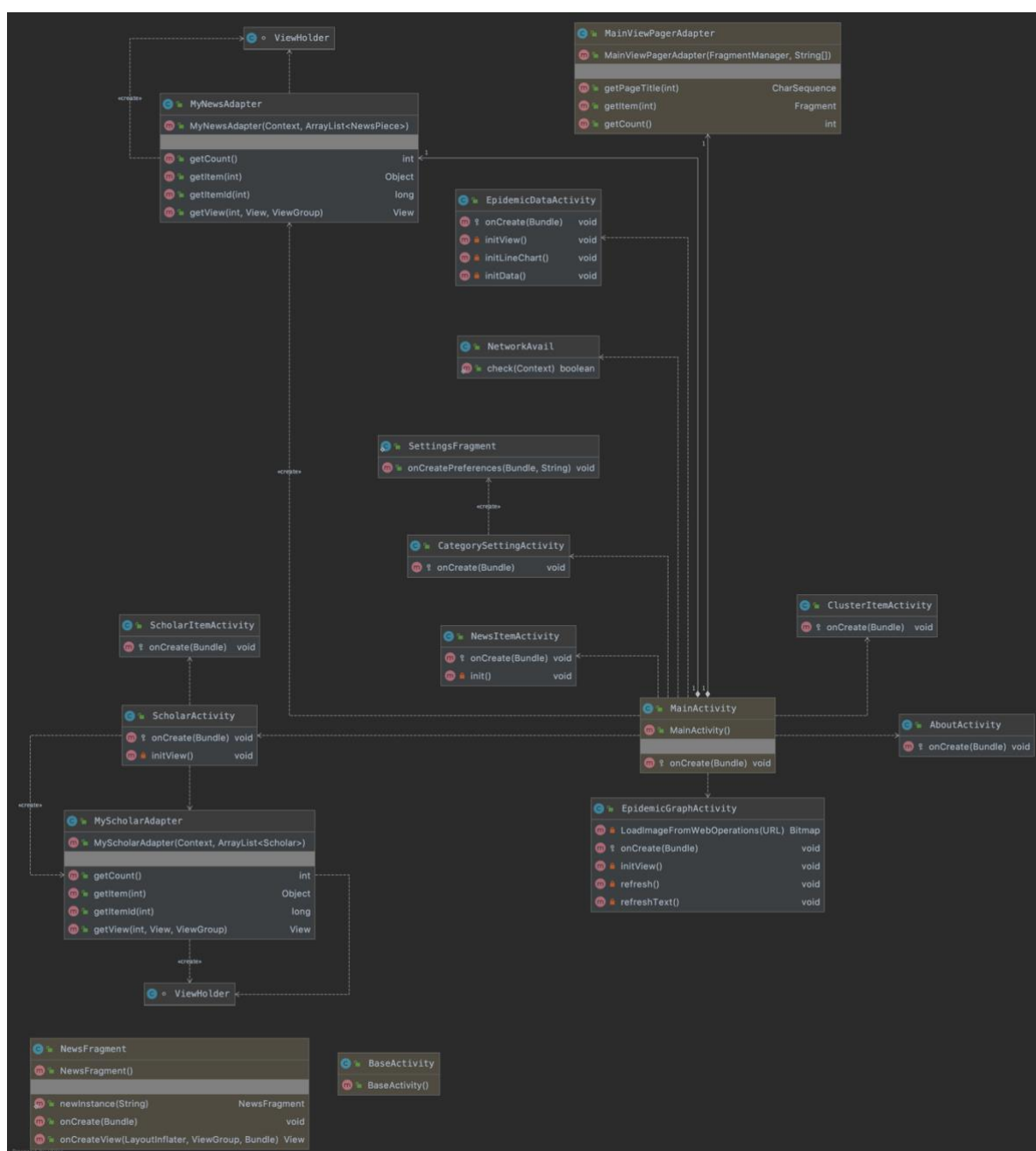


`backend.news` 包实现疫情新闻相关后端, `NewsPiece` 类定义一条新闻, `NewsHistory` 类定义新闻历史记录, `NewsAPI` 是供前端调用的 API; `backend.cluster` 包实现聚类相关后端, `Event` 类定义了用作聚类的一个新闻事件, 继承 `NewsPiece`, `ClusterAPI` 是供前端调用的 API, 聚类数据由 `python` 代码生成; `backend.kg` 包实现疫情图谱相关后端, `Entity` 类定义了疫情图谱中的一个条目, `GraphAPI` 是供前端调用的 API; `backend.scholar` 实现知疫学者相关后端, `Scholar` 类定义一个学者, `ScholarAPI` 是供前端调用的 API; `backend.stat` 实现疫情统计数据相关后端, `EpidemicStat` 定义某个地区(国家/省/城市)的统计数据, `StatAPI` 是供前端调用的 API; `backend.BackEnd` 定义了。以上均由谢云桐实现。

前端方面代码结构 UML 图如下：



MainActivity 为程序的入口，主要用于展示新闻列表以及导航栏、菜单栏的实现。NewsItemActivity 用于展示具体新闻，MyNewsAdapter 为用于展示新闻列表的适配器。CategorySettingActivity 用来添加删除分类操作。EpidemicDataActivity 用来展示疫情数据的折线图，EpidemicGraphActivity 用来展示疫情图谱，ClusterItemActivity 用来展示聚类具体新闻，ScholarActivity 用来展示知疫学者列表，ScholarItemActivity 用来展示具体学者，MyScholarAdapter 为用于展示学者列表的适配器。AboutActivity 用来展示关于。NetworkAvail 用来判断当前网络是否可用。其中 CategorySettingActivity 由谢云桐实现，其余均为黄翘楚实现。

具体代码层面的分工情况，可以参考 <https://github.com/xieyt2000/XieHuangNews> 的相关记录。

2 具体实现

	功能	描述	分工
基础功能	系统支持	正常运行，不崩溃不卡顿	谢云桐、黄翹楚
	页面布局	布局合理、美观，点击反馈	绝大部分由黄翹楚完成，小部分由谢云桐完成
	分类列表	删除添加分类，动态特效	谢云桐
	新闻列表	显示新闻列表，布局美观，新闻详情页	前端黄翹楚，后端谢云桐
		新闻的本地存储，离线浏览，已读标灰	前端黄翹楚，后端谢云桐
		上拉获取更多，下拉刷新，带有特效	前端黄翹楚，后端谢云桐
		显示新闻的来源和时间，布局美观	前端黄翹楚，后端谢云桐
		新闻关键词搜索、历史记录，页面美观	前端黄翹楚，后端谢云桐
	分享收藏	分享新闻摘要到常用 App	谢云桐
	疫情数据	全国各省、全球各国数据统计折线图	前端黄翹楚、后端谢云桐
拓展功能	疫情图谱	查询疫情图谱内容，展示描述、属性和关系	前端黄翹楚，后端谢云桐
	疫情新闻聚类	对疫情相关新闻进行聚类并展示，页面美观	前端黄翹楚，后端谢云桐
其他拓展	知疫学者	显示高关注度学者和追忆学者，查看详情，页面美观	前端黄翹楚，后端谢云桐
	清除缓存	删除历史记录等本地缓存	前端黄翹楚，后端谢云桐
后期工作	实验报告	包括本报告和 <code>readme.md</code>	谢云桐
	演示视频		录制黄翹楚，剪辑谢云桐

谢云桐：我工作的难点在于网络线程的处理。安卓为了防止网络访问阻塞 App，因此禁止在主线程访问互联网，必须新开线程，这考验了我们对多线程的运用能力。为了优化操作逻辑与用户体验，本程序必须不同的模块分别使用同步请求和异步请求。比如查看学者详情时需要下载一个图片，可能会耗时比较长，提前将所有图片缓存下来又会造成过多负担，因此只能选择同步请求；而疫情数据、学者简介（仅文字）等体量没有这么大的数据则是使用异步请求获取，即在程序开始时新开线程获取，需要使用时无需访问网络、阻塞主线程。由于 App 最开始载入时需要获得这些数据，所以如果打开 App 时网络未连接，可能无法使用部分功能（离线功能一定可以正常使用）。

在实现分享功能时，我也遇到了一些困难。我们提交版本的分享功能非常简单，就是调用安卓机器自带的分享，分享一段纯文本的新闻摘要，但是最初我们尝试使用 ShareSDK，并在微信、微博等平台申请分享权限，然而如下图所示，在所有平台都因官方网站（提交的 github 项目地址）问题被驳回，最后不得不放弃。在这种情况下使用 ShareSDK 的效果就不如自带的分享了，于是只能全盘抛弃之前的工作。

关于聚类功能，我直接使用 python 实现，调用了 sklearn 库里的 tf-idf 和 k-means，k-

means 算法复杂度为 $O(knT)$ ，其中 k 表示分类数目，我们分为 10 类，即 $k=10$ ， n 表示数据个数， T 表示迭代次数。



黄翹楚：我工作的难点主要在于前端的布局与美化，由于之前从未接触过安卓开发，对前端各种组件以及其各种属性并不熟悉，导致在开发的时候踩了很多坑，不同组件不同初始化顺序会有不同的结果，有时甚至会导致程序崩溃，各个组件的大小、排布、颜色也需要经过多次调试才能确定。不过随着对各个组件的掌握越来越熟练，开发速度也越来越快。当然在开发的过程中也用到了前人已经写好的包，上网查询各种组件美化，这大大降低了工作量，同时使得界面更加美观。

3 总结与心得

今年的形势与往年大不相同，受疫情影响，我们上学期有两门课（《人工智能导论》和《形式语言与自动机》）都将期末考试安排到了小学期的第三周末和第四周初，导致我们在这之后才开始开发，时间上要紧张许多，不过双人合作的模式也确实减轻了我们的负担，因此最后还是可以按时完成作业，达到较好的效果。

在这两周的时间里，我们学习了安卓的前后端开发，熟练了 Java 的使用，同时锻炼了合作完成项目的能力，可谓是收获颇丰。不过感觉我们在开发之前有关安卓编程实在是所知甚少，毕竟只上了两节有关安卓开发的课，甚至没有布置过安卓的小作业，希望助教和老师以后可以优化一下课程结构，加大安卓开发的比重。