Detection and Prevention of Android Ransomware



方案

1. 数据集介绍

我们使用通过翻译库翻译一部分中文内容（抓取和手动获得的）汉-> 英，

情绪标签进行标记，然后混合开源几个数据集混合（主要英文为主）组成我们自有数据csv（环球网各种频道、本地新浪国内新闻、各种资讯），然后丢进我们翻译库进行翻译，然后合并到我们以前的综合情绪数据集。

翻译库用的是：<https://www.cnpython.com/pypi/argostranslate> .

该translator 对象获取从源语言到目标语言的句子列表。语言是两个字母的 ISO 639 代码，上面我们创建了一个从俄语 (ru) 到英语 (en) 的翻译器。这将获得一个模型和一个标记器。 它支持非常多的语言列表，您可以在 Hugging Face 的模型中心查看所有模型。它可以翻译的文本长度、运行速度以及它如何响应不寻常的标点符号（相对于它所训练的数据）。

我们的爬虫部分主要是用：scrapy+selenium框架图像

上图主要是爬虫部分的架构图。

引擎从调度器中取出一个链接(URL)用于接下来的抓取

引擎把URL封装成一个请求(Request)传给下载器

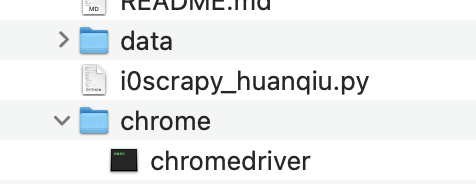
下载器把资源下载下来，并封装成应答包(Response)

爬虫解析Response

解析出实体（Item）,则交给实体管道进行进一步的处理

解析出的是链接（URL）,则把URL交给调度器等待抓取

如果遇到反爬措施比较严格的网站，我们使用selenium进行模拟浏览器。



利用浏览器原生 API，封装成一套更加面向对象的 Selenium WebDriver API，直接操作浏览器页面里的元素，甚至操作浏览器本身（截屏、窗口大小、启动、关闭、安装插件、配置证书...）

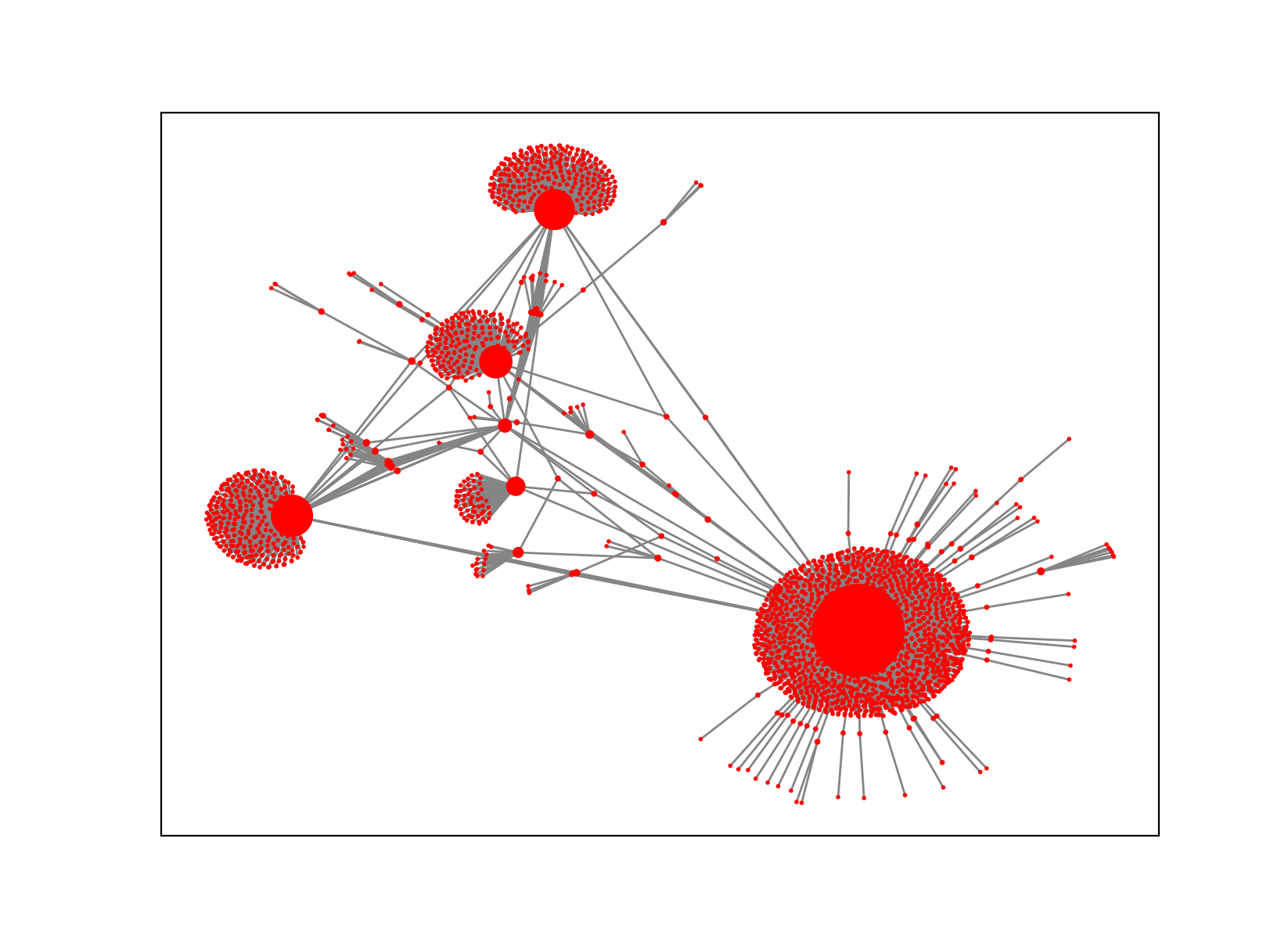
* webdriver 本质是一个 web-server，对外提供 webapi，其中封装了浏览器的各种功能图像
* 不同的浏览器使用各自不同的 webdriver（由浏览器厂商提供）

selenium 还可以处理一部分带验证码和点击的情况。

我们的数据集主体还是使用啦twitter/quora数据集有：关于美国各大航空公司问题的情绪分析工作。 Twitter 数据是从 2015 年 2开始收集的，并要求贡献者首先对正面、负面和中立的推文进行分类，然后对负面原因进行分类（例如“迟到的航班”或“粗鲁的服务”）。python脚本预处理步骤对收集的数据集执行必要的数据预处理和清理。 在先前收集的数据集上，有一些关键属性文本：tweet本身的文本，created\_at：创建日期，favorite\_count，retweet\_count：收藏和转推的数量，已收藏，已转推：布尔值，指明是否通过身份验证的用户(您)对此推文等有帮助或转发。我们已应用了广泛的预处理步骤，以减小功能集的大小，使其适合于学习算法。 清洁方法基于字典方法。

从quora/Twitter获得的数据通常包含许多HTML实体，例如＆lt; ＆gt; ＆amp; 嵌入到原始数据中。 因此有必要摆脱这些实体。 一种方法是通过使用特定的正则表达式直接删除它们。 野兔，我们正在使用PythonHTML解析器模块，该模块可以将这些实体转换为标准HTML标记。 例如＆lt; 转换为“ <”和＆amp; 转换为“＆”。 此后，我们将删除此特殊HTML字符和链接。 在解码数据时，这是将信息从复杂的符号转换为简单易懂的字符的过程。 收集的数据使用不同的解码形式，例如“拉丁”，“ UTF8”等。

我们在进行ML训练前还进行啦数据探索。分析总体数据集（twitter+quora+自有标记数据来源之间的关联度分析）



随着大数据时代的到来，人们积累数据的意识越来越强，如何从积累的数据中获取我们想要的信息越来越受到关注。社交网络、twitter或facebook上的公众评论、公司电子邮件记录、疾病发展、网站访问等。越来越多的可以连接到网络的数据被保存下来，因此学习分析这些数据可以帮助我们找到信息我们需要在凌乱的连接关系中

我们首先使用networkX抓取推特的数据，得到推特2的关键数据。结果表明，网络密度为0.0904，说明网络中的参与者具有一定的相关性。网络中总共有所有节点。共有3287条线，节点之间的最长距离为6，即任意两个节点之间的最长距离为6条线。距离反映了演员之间间隔的长短，距离越小，说明彼此的联系越紧密。也就是说，在推特上，媒体机构之间最多可以通过5个节点相互连接，这比世界上任何两个人之间的平均距离，即距离“六度分离”只有5个节点的距离还要小。理论强调。此外，网络节点之间的平均距离为2.937，这意味着任何两个媒体机构只需要通过1到2个节点的中介即可成为网络中的互联节点。我们的数据集没有特别的富集，比较符合六度原理：（有大v，也有普通网民，之间通过社交网络和进行啦关联，不存在大规模数据评论情绪被操控的情况）。不存在数据偏见，我们可以进行上述清洗、合并之后进行训练。

我们进行python脚本中数据分析，在i2deep-learning-for-sentiment-analysis.ipynb中拆除需要的列，进行各种模型适配。

2. 做一个后端**api**服务，在一个后端**webserver**部署起来（**wsgiserver/flask api**等）

--- 整体的流程 ----

我们的前后端主要基于：flask+sqlchemy + numpy+ html5+vue+jquery技术栈

先把第一步训练好的机器学习模型和参数缓存起来，然后放在flask同级别地方方便调用。

对于科学数据集的管理：是flask 后端+ flask cros ,前端vue+bootstrap+ jinja 接受之后，传递到后端sqlchemy进行存储

整体上主要是从前端Jinja的template页面触发事件，然后通过Flask API传递到android 预测client调用

Flask 是一个基于 Python 的轻量级 Web 框架，WSGI 工具箱采用 Werkzeug，模板引擎使用 Jinja2。由于其不依赖于特殊的工具或库，并且没有数据抽象层、表单验证或是其他任何已有多种库可以胜任的功能，从而保持核心简单、易于扩展，而被定义为"微"框架。但是，Flask 可以通过扩展来添加应用功能。并且 Flask 具有自带开发用服务器和 debugger、集成单元测试和 RESTful 请求调度 (request dispatching)、支持 secure cookie 的特点。我们就主要使用Flask的网站部分和wsgi写API部分

# --- 分个介绍 ----

Jinja2 是基于 Python 的模版引擎，支持 Unicode，具有集成的沙箱执行环境并支持选择自动转义。Jinja2 拥有强大的自动 HTML 转移系统，可以有效的阻止跨站脚本攻击；通过模版继承机制，对所有模版使用相似布局；通过在第一次加载时将源码转化为 Python 字节码从而加快模版执行时间。我们的网站看得到的页面部分是这块开发。

Flask

1、Flask主要包括Werkzeug和Jinja2两个核心函数库，他们分别负责阢处理和安全方面的工鞥呢，这些基础函数为Web项目开发过程提供了丰富的基础组件。

　　2、Flask中的Jinja2模板引擎，提高了前端代码的复用率。可以大大提高开发效率并且有利于后期的开发与维护。

　　3、Flask不会指定数据库和模板引擎等对象，用户可以根据需要自己选择各种数据库。

　　4、Flask不提供表单验证功能，在项目实施过程中可以自由配置，从而为应用程序开发提供数据库抽象层基础组件，支持进行表单数据合法性验证、文件上传处理、用户身份认证和数据库集成等功能。

Flask的特点可以概括为：因为灵活，轻便高效，被业界所认可，同时拥有基于Werkzeug、Jinja2等一些开源库，拥有内置服务器和单元测试，适配RESTful。我们使用flask编写网站的用户登录/注册/权限管理/个人主页/机器学习训练和可视化的前后台逻辑部分，非常方便后续进行扩展。

我将使用 SQLite，这是一个小型 SQL 数据库实现，非常容易启动和运行。请记住，您可能想在生产环境中考虑更可靠的数据库，例如 PostgreSQL 或 MySQL。图像

flask\_sqlalchemy

要在 Flask 项目中设置 SQLAlchemy，我们可以导入 flask\_sqlalchemy 软件包（我们之前已安装），然后将 Flask app 变量包装在新的 SQLAlchemy 对象。我们还希望在 Flask 应用程序配置中设置 SQLALCHEMY\_DATABASE\_URI 以指定我们要使用的数据库以及如何访问它

# -- api 风格---

最后，我们可以开始定义 RESTful 处理程序。我们将使用 Flask-RESTful 软件包，这是一组工具，可帮助我们使用面向对象的设计来构建 RESTful 路由。

REST架构风格

六条设计规范定义了一个 REST 系统的特点:

客户端-服务器: 客户端和服务器之间隔离，服务器提供服务，客户端进行消费。

无状态: 从客户端到服务器的每个请求都必须包含理解请求所必需的信息。换句话说， 服务器不会存储客户端上一次请求的信息用来给下一次使用。

可缓存: 服务器必须明示客户端请求能否缓存。

分层系统: 客户端和服务器之间的通信应该以一种标准的方式，就是中间层代替服务器做出响应的时候，客户端不需要做任何变动。

统一的接口: 服务器和客户端的通信方法必须是统一的。

按需编码: 服务器可以提供可执行代码或脚本，为客户端在它们的环境中执行。这个约束是唯一一个是可选的。

Flask-RESTful

我们需要设置 Flask-RESTful 扩展名才能在 Flask 服务器中启动并运行。Flask-RESTful 是一个 Flask 扩展，它添加了快速构建 REST APIs 的支持。它当然也是一个能够跟你现有的ORM/库协同工作的轻量级的扩展。Flask-RESTful 鼓励以最小设置的最佳实践

Vue

整个平台的前端部分和可视化部分我们主要是使用vue+jquery+html5: Vue 是一套用于构建用户界面的渐进式 JavaScript 框架 ；同时它是一个典型的 MVVM 模型的框架（即：视图层-视图模型层-模型层）;HTML5是HTML的新标准，是一种超文本标记语言，是用来创建网页的标准标记语言，通过一系列的标识，来规范网络上的文档格式;区别：

        1.vue是一个渐进式 JavaScript 框架，而HTML5是一种超文本标记语言 2.在开发中vue框架通过mvvm的模式，解耦了视图层与模型层，而HTML5原生开中数据与标签紧耦合；    但是vue和html5可以进行结合:    vue是一个前端框架，但还是建立在HTML ，CSS ，JavaScript的基础之上的，通过编译之后依然是HTML+CSS+JavaScript组成。

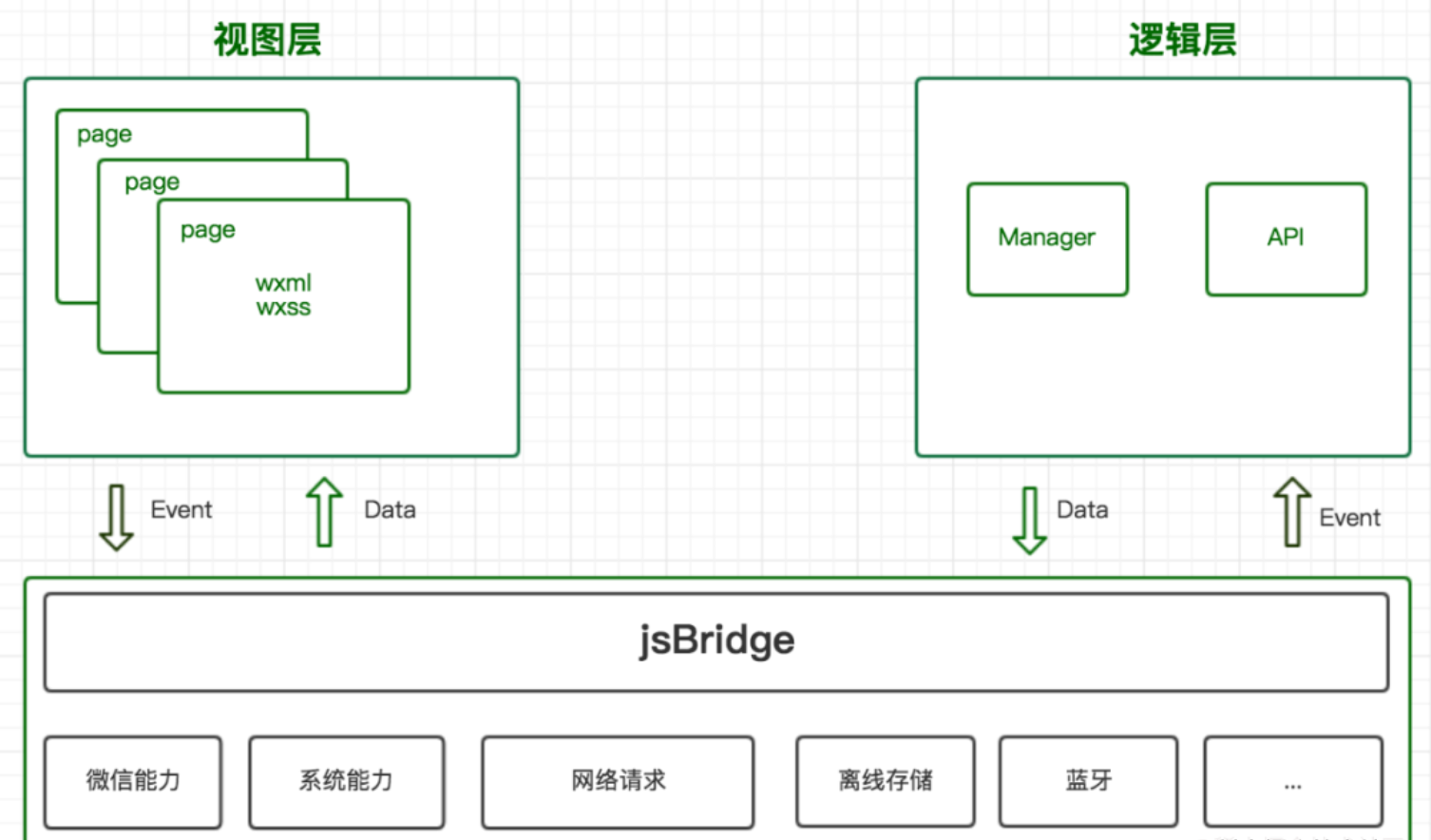
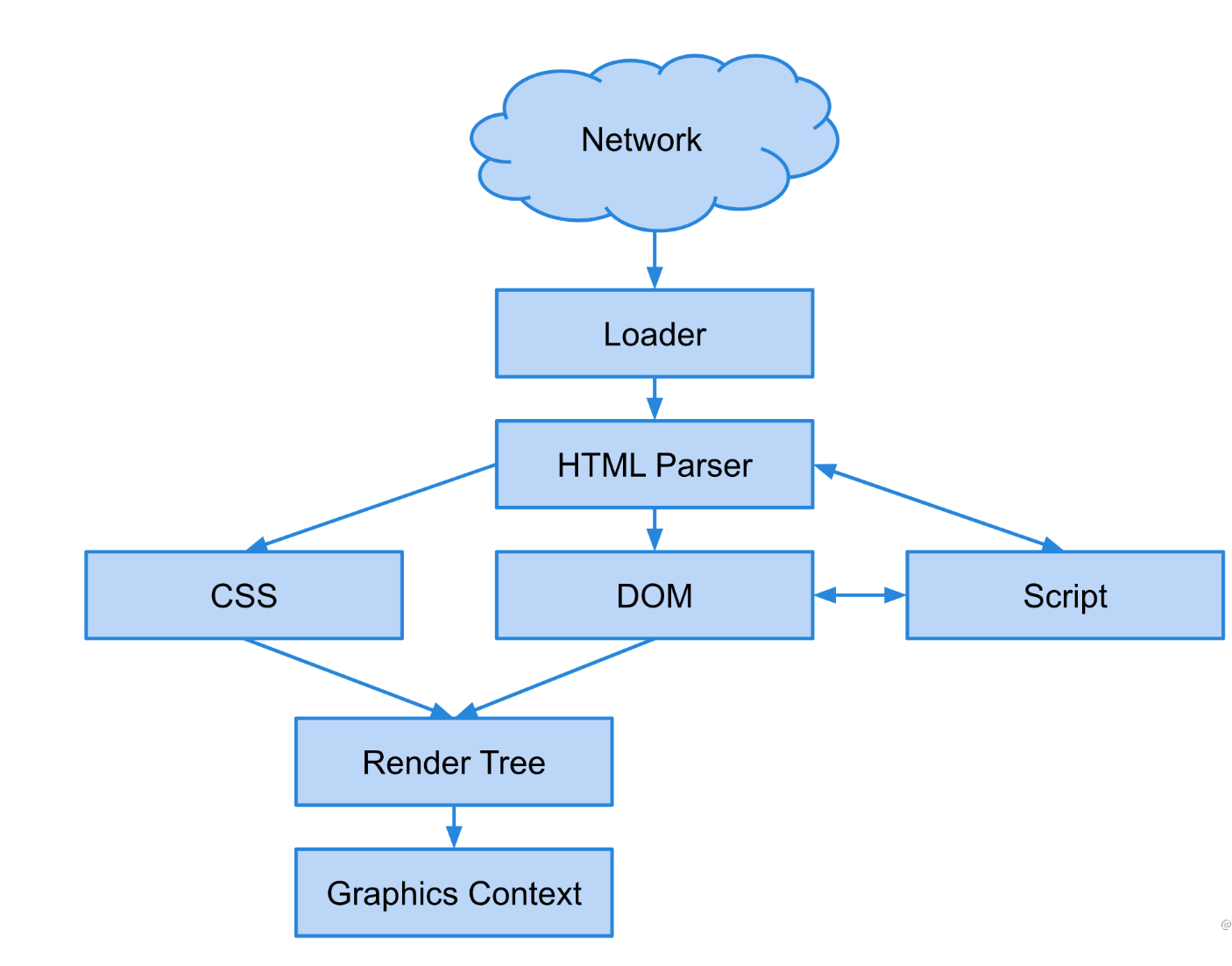
3. 把第一步的机器学习模型训练好的模型缓存起来，拷贝到 **api**后端工程，集成在一起，提供接受输入，

模型进行分类预测的**api**

4. **android**手机端

/https://stackoverflow.com/questions/30173080/how-can-i-analyse-a-apk-file-in-android-application-get-the-permissions

写一个android app 即成这个overflow说的库，然后调用库分析某些本地库权限list/或者是分析api调用序列（看第一步使用的是什么数据集而定）

然后app端根据获取的权限，动态传回机器学习api接口，然后获得预测结果，进行不同的弹框提示

通过不停的轮巡后端i4security\_predict\_web中的api，和已经适配啦移动端的webpage （通过bootstrap适配）。现 Native端和 web 端双向通信的一种机制。在android4.0之后，andriod studio在Hybrid开发模式下，H5页面经常需要使用到Native的功能，比如打开二维码扫描、调用本地相册、获取用户信息等，同时Native也需要向Web端发送推送、更新状态等。webview容器在原生中是一个容器控件，可以加载url或者 web页面，而JavaScript是运行在单独的JS Context中（Webview容器、JSCore等），与原生有运行环境的隔离，所以需要有一种机制实现Native端和Web端的双向通信。

在android中，如果后端爬虫爬去更新数据，然后通过调用我们训练好的情绪分析及其模型，之后存入sqlchemy对应的数据库，然后我们的api接口会实时变化和更新，android调用的js+html5网页可以实时获得新数据的通知。