2.2 基于 SOA 的开发方式

面向服务的体系结构 (Service-Oriented Architecture, SOA), 是一个组件模型, 它将应用程序的不同功能单元 (称为服务) 通过这些服务之间定义良好的接口和契约联系起来. 通过 SOA 的设计模式, 可以将 D-M-V 的三层结构中每层结构分离开来, 降低系统耦合程度, 将系统入口统一到 Web 页面进行管理. 使用这样的设计模式, 可以将系统的不同组件 (例如数据存储, 数据建模和数据可视化) 分发到不同的计算机中. 基于分布和松散耦合的特性, 分析任务可以利用并行计算资源统一的服务接口使得用户使用系统更加灵活, 基于 Web 的形式, 将复杂的原始数据、多样的模型和操作系统与用户隔离开来, 同时, 使用基于 Web 的形式更有利于集成其他基于 Web 的服务, 如风险地图等.

2.3 基于 TextCNN 的短文本分类算法

TextCNN 是 Yoon Kim 在 2014 年提出的一种用于文本分类的算法[1], 主要思路是将 CNN (Convolutional Neural Networks) 的技术用于文本分类, 通过利用多个不同大小的卷积核 (kernel) 来提取相关句子里的关键信息 (与多窗口大小的 n-gram 模型类似), 从而可以更好地捕捉文本的局部相关性. 网络结构主要包括输入层 (input layer)、嵌入层 (embedding layer)、卷基层 (convolutional layer)、池化层 (pooling layer) 和输入层 (output layer).

在本平台的实验数据集上, 使用 TextCNN 对含有食源性关键字的微博进行事件探测, 可以达到最好的分类效果.

2.4 模型自适应选择

模型的自适应选择主要包括两个部分, 包括对于特征的选择和对于预测模型的辅助选择, 本文通过对 Xgboost 对特征的贡献进行计算, 减去无效的特征, 对于模型的选择, 系统提供多种评价指标, 包括 AUC(Area Under Curve), 即受试者工作特征曲线 (ROC) 下方的面积, 精准率 (Precision), 召回率 (Recall), F1 值 (F1 Score) 等作为模型的评价指标, 辅助用户进行模型选择.

3 系统实现

3.1 分布式爬虫

面向多源网络数据, 平台集成了分布式爬虫系统, 并基于分布式爬虫系统开发了多种爬虫程序, 能够自动抓取社交媒体数据, 如微博、美团网评论数据, 出行数据如共享单车出行数据, 官方统计数据, 如国家统计局相关统计数据.

自动数据收集是整个系统的基础. 具体包括自动数据采集和原始数据的存储. 分析所需的数据可以分为静态数据和动态数据两类. 静态数据是主要在官方网站上发布的数据更新频率较低或经常更新的政府或机构. 静态数据提供基本信息, 包括数字地图和其他地理数据, 气候数据, 记录环境特征的遥感数据和社会经济统计数据. 动态数据是从动态更新的网络媒体和社交网络收集的数据. 微博、美团网评论数据被设置为自动爬虫的目标. 为此我们为不同的数据源部署自动爬虫, 完全不同的数据结构, 并保持最新信息被及时检测和存储.

静态数据源提供具有明确地理坐标或位置的结构良好的数据. 官方网站通常会定期发布数据. 为了接收这些具有可控开销的静态数据, 收集静态数据的搜索器被分配定时任务以检查网站的更新并请求最新发布的数据. 由于结构不变, 地理标签明确, 所收集的数据在存储之前不需要太复杂的处理. 对于动态数据, 自动收集更复杂. 为了获得实时的信息, 爬虫必须不断地监控新闻和社交媒体的网页, 寻找特殊的关键词, 包括突发事件的描述, 疾病的名称等, 这些都可以看作是突发事件的标志.

本文开发的分布式爬虫系统采用 Celery (http://www.celeryproject.org/) 作为分布式任务队列, 使用 Rides 作为分布式后端, 使用 MongoDB (https://www.mongodb.com/) 作为分布式数据库, 基于 Python requests (http://www.python-requests.org/) 发送网络请求及下载页面, 使用 beautiful soup4 (https://www.crummy.com/software/BeautifulSoup/) 解析页面, 基于 Flask (http://flask.pocoo.org/) 开发 web 界面管理整个数据采集系统. 系统结构如图 3 所示.

社交媒体网站如微博、美团网等对于爬虫往往有着严格的限制. 为此, 爬虫系统建立了 IP 池、账号池解决限制访问的问题, 对于验证码则通过 CNN 进行验证码识别, 多次识别错误时接入人工打码平台进行验证码识别.

3.2 多源数据融合

由于影响食源性疾病的因素众多, 本文采用了多种数据, 数据格式与来源见表 1. 面对多源异构数据, 首先要做的是对多源异构数据进行数据融合, 置于统一

的时空坐标系中. 本文所用数据虽然来源众多, 但主要有四种格式, 栅格数据、矢量数据、表结构化数据以及文本数据. 对于栅格数据和矢量数据主要问题在于不同的数据采用的投影坐标系以及地理坐标系都不尽相同, 首先要将其置于统一的时空坐标系当中. 本文使用 Proj4 (https://proj4.org/) 进行投影坐标转换, GDAL 库 (https://www.gdal.org/) 进行矢量、栅格数据的提取.

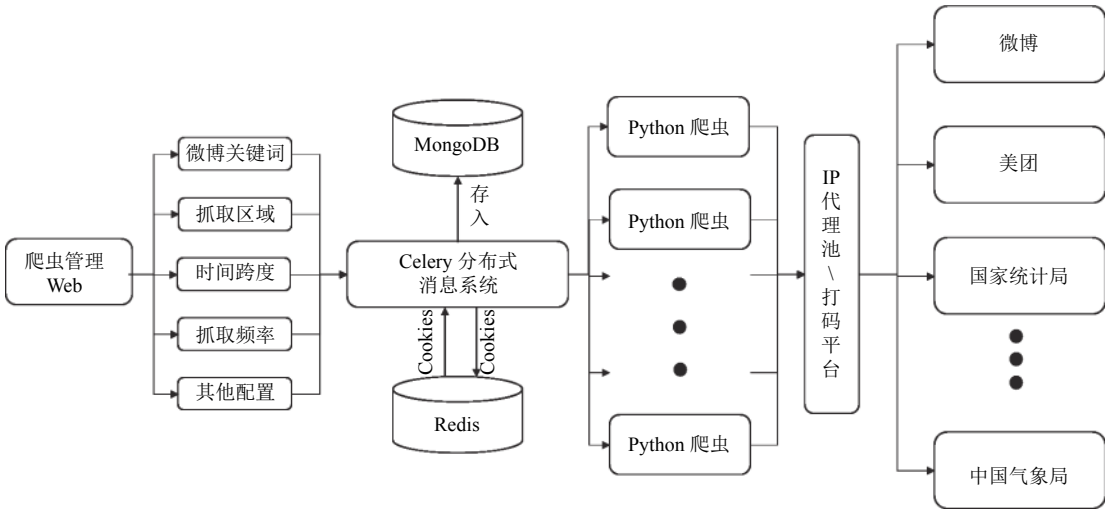


图 3 分布式爬虫系统

表 1 多源数据类型与来源

数据类别	数据描述	数据格式	数据来源
环境数据	温度	栅格数据	中国气象数据网
	降水量	栅格数据	中国气象数据网
	空气质量 (AQI)	表格结构化数据	中国环保部
	植被系数 (NDVI)	栅格数据	http://www.resdc.cn/
	海拔高度 (DEM)	栅格数据	http://www.gebco.net
社会经济数据	GDP	栅格数据	http://www.resdc.cn/
	人口数据	栅格数据	http://www.resdc.cn/
社交媒体数据	微博	文本数据	微博
	美团网评论数据	文本数据	美团网
交流数据	商铺数据	表格结构化数据	美团点评
	医院数据	矢量数据	OpenStreetMap
	道路数据	矢量数据	OpenStreetMap
	兴趣点数据	矢量数据	OpenStreetMap
出行数据	共享单车 OD 矩阵	表格结构化数据	摩拜单车 APP

由于直接通过 gdal 对时空数据进行读取比较低效的, 为此, 在提取时空坐标上的不同属性信息后, 基于空间数据的最小粒度, 建立时空数据库, 以时空立方体的形式以进行高效索引.

本文采用的文本数据主要是微博语料数据、美团网评论语料数据, 对于事件时间, 可以抓取到微博发博时间, 评论发送时间来确定, 地理坐标则根据下文的地理坐标推断算法进行推断. 结构化数据本身已经包含了时空信息, 可以直接进行使用.

3.3 事件探测与关键信息推断

3.3.1 基于短文本分类的时间探测算法

微博等社交媒体数据蕴含了大量的信息, 然而由于其本身的特点, 其中的噪声也特别多. 为了充分利用微博数据进行食源性疾病的检测, 首先要对数据进行清洗. 为了去除干扰的僵尸微博账户, 本文目前利用用户的关注粉丝比、微博总量等作为筛选条件, 选出真正有价值的微博内容, 同时基于短文本分类算法开发了食源性疾病事件探测算法, 对微博语料中的食源性

疾病进行探测.

常见的短文本分类算法流程如图 4 所示, 主要包括分词、去停用词、词向量的训练以及分类器的训练. 传统的短文本分类算法主要是根据分词后计算的词组 TF-IDF 权重, 然后使用朴素贝叶斯分类器进行分类. 本文开发的平台采用 Jieba (https://github.com/fxsjy/jieba) 分词库进行分词, 基于 sklearn (http://scikit-learn.org/) 开发 TF-IDF 算法, 基于 gensim (https://radimrehurek.com/gensim/) 开发 Word2vec[12]算法, 基于 TensorFlow (https://tensorflow.google.cn/) 开发 Fasttext[13]和 Textcnn 算法, 平台集成多种分类算法, 并且展现分类效果, 供用户选择.

图 4 短文本分类流程

3.3.2 基于动态上下文的地理位置推断算法

由于新浪微博每条所含字数小于 140 个字, 平均微博长度为 30 个字左右, 一条微博很难全面准确的描述食品安全事件. 由于用户很有可能会有连续多条微博涉及食源性疾病问题, 而其中只有某一条直接含有食源性疾病关键词, 其他相关微博可能含有有关食源性疾病的其他重要信息, 如地理位置等. 那么, 简单地根据关键词筛选单条微博的方法会错过许多含有重要信息的微博. 本文目前研究采用动态上下文确定事件窗口, 根据事件窗口确定候选微博. 动态上下文窗口, 是依据微博之间的语义相似度来确定的, 分别向前、向后利用微博间的文本相似性来确定上下文窗口.

地理位置推断算法流程如图 5 所示, 对于美团网评论数据, 根据店铺名称获得地理位置坐标. 对于微博数据, 首先对微博文本检测是否包含地理位置名词, 若包含地理位置名字, 则根据地理名称的 Geocoding 获取地理坐标, 若单条微博文本中不包含地理位置信息, 则在候选微博集中寻找地理位置信息, 若微博上下文中也未包含地理位置信息, 则根据用户注册地址确定食源性疾病事件地理位置.

3.4 基于多源数据的风险预测算法

在多源数据的基础上, 经过丰富的特征工程, 开发了食源性疾病的风险预测算法, 使用多种机器学习算法, 开发了多种风险预测模型, 如逻辑回归、决策树、梯度提升树, 随机森林等, 并且提供多种的评测标准. 并将多种模型集成在大数据处理平台上, 使用并行化算法优化模型, 使得平台能够利用多源大数据快速计算食源性疾病爆发的区域风险.

3.5 可视化与交互系统

对于模型的结果的展示, 本文基于 Echarts (https://echarts.baidu.com/) 实现基于地图的风险可视化, 交互系统采用 Web 形式, 基于 Flask 开发 Web 后台, 前端采用 bootstrap UI 以 JavaScript 进行开发.

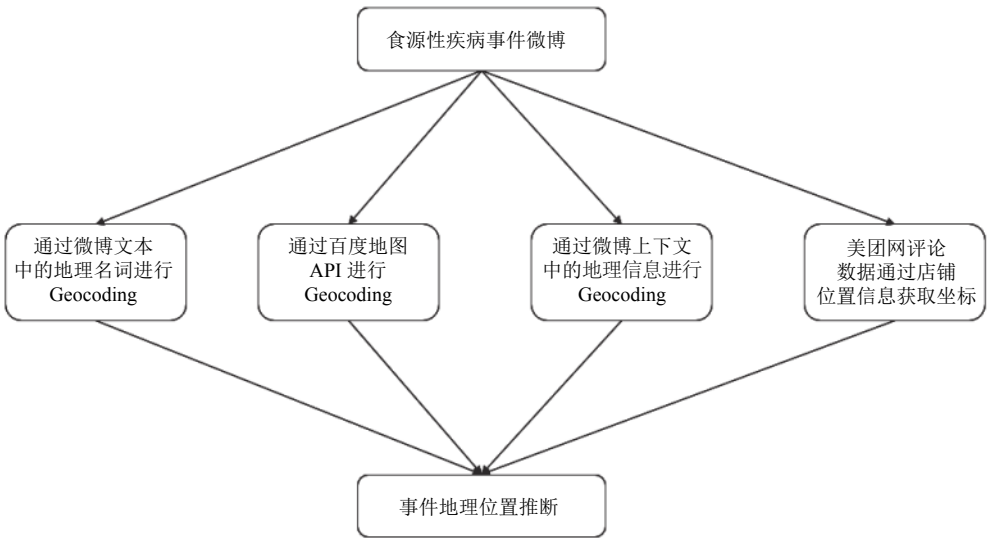
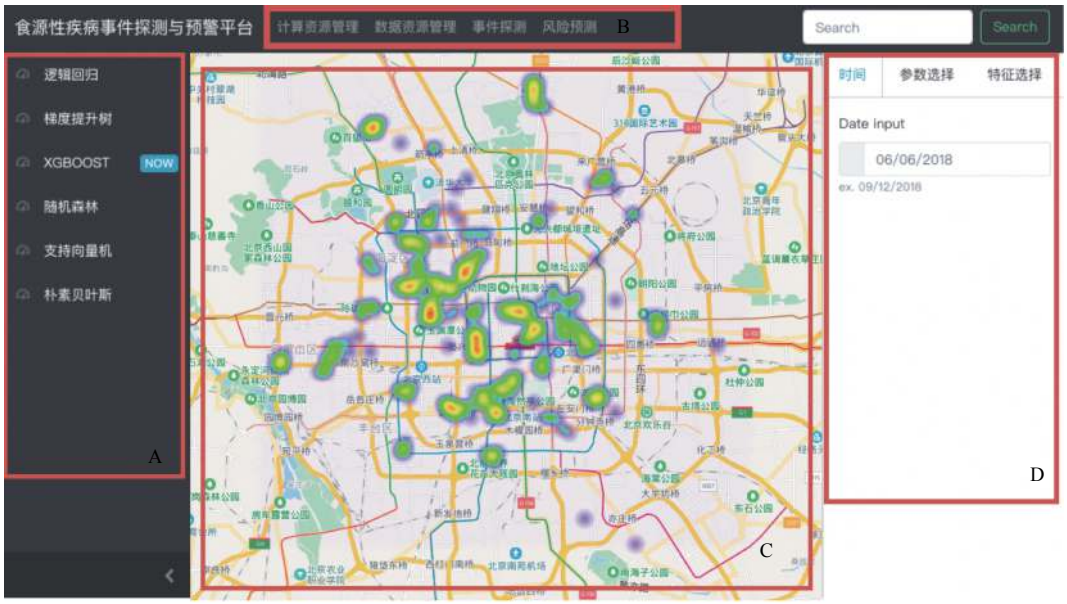


图 5 食源性疾病事件地理位置推断

4 平台验证与试验结果

4.1 系统界面及风险预测示意图

通过平台抓取北京市相关数据, 获取食源性疾病相关微博 10 万条, 美团网评论数据 500 万条, 结合环境数据、官方统计数据等多源数据, 数据来源与格式见表 1, 系统界面及预测风险如图 6 所示.



A 区域为模型选择区，B 区域为导航栏，跳转至计算机、数据管理等界面，
C 区域为基于地图实现的风险地图，D 区域为模型的参数与特征选择

图 6 2018 年 6 月食源性疾病风险示意图

4.2 事件探测算法实验结果

本文通过人工标注了的 3 万条包含食源性疾病关键字的微博进行训练, 训练集和测试集的比例为 8:2. 通过 AUC 对分类结果进行评价. 实验结果见表 2.

表 2 事件探测算法实验结果

模型名称	AUC
TF-IDF+Bayes	0.77
Word2Vec+LR	0.79
Word2Vec+GBDT	0.81
Fasttext	0.84
TextCnn	0.86

从实验结果可以看出, 使用 TextCnn 文本分类算法可以达到最高的分类准确度, 能够较好的识别出短文本中的食源性疾病事件.

4.3 风险预测算法实验结果

根据社交媒体中的病例信息, 通过上下文地理位置推断算法推断出地理位置后, 使用多种模型进行风险预测, 在小区域粒度下, 预测在给定时间条件下, 一个地点是否会发生食源性疾病, 通过 AUC 进行评价, 实验结果见表 3.

表 3 风险预测算法实验结果

模型名称	AUC
Logistic Regression	0.73
Gradient Boosting Decision Tree	0.85
XGBOOST	0.86
Random Forest	0.83
Support Vector Machine	0.79

从实验结果可以看出, 使用 XGBOOST[14]对北京食源性疾病发生的风险进行预测可以达到最好效果.

5 结论与展望

本文使用大数据与人工智能的方法对食源性疾病事件进行探测和风险预警. 面向食源性疾病的数据获取、数据分析和数据可视化的需求开发了食源性疾病事件智能探测与预警平台, 使用多源数据对食源性疾病事件的风险进行预测与评估, 为食源性疾病的管理与防治提供一定的指导作用.

参考文献

1 Manyika J, Chui M, Brown B, *et al*. Big data: The next

frontier for innovation, competition, and productivity. 2011. http://www.mckinsey.com/insights/business_technology/big_data_the_next_frontier_for_innovation.

2 Carneiro HA, Mylonakis E. Google trends: A web-based tool for real-time surveillance of disease outbreaks. Clinical Infectious Diseases, 2009, 49(10): 1557–1564. [doi: 10.1086/599193]

3 Yuan QY, Nsoesie EO, Lv BF, *et al*. Monitoring influenza epidemics in china with search query from baidu. PLoS One, 2013, 8(5): e64323. [doi: 10.1371/journal.pone.0064323]

4 陈翔, 徐佳, 吴敏, 等. 基于社会行为分析的群智感知数据收集研究. 计算机应用研究, 2015, 32(12): 3534–3541. [doi: 10.3969/j.issn.1001-3695.2015.12.003]

5 Harris JK, Mansour R, Choucair B, *et al*. Health department use of social media to identify foodborne illness—chicago, illinois, 2013-2014. Morbidity and Mortality Weekly Report, 2014, 63(32): 681–685.

6 郭旦怀, 崔文娟, 郭云昌, 等. 基于大数据的食源性疾病事件探测与风险评估. 系统工程理论与实践, 2015, 35(10): 2523–2530. [doi: 10.12011/1000-6788(2015)10-2523]

7 Chandra S, Khan L, Muhaya FB. Estimating twitter user location using social interactions--a content based approach. Proceedings of 2011 IEEE 3rd International Conference on Privacy, Security, Risk and Trust and 2011 IEEE 3rd International Conference on Social Computing. Boston, MA, USA. 2011. 838–843

8 祝天刚, 郭旦怀, 王学志, 等. 基于短文本的食源性疾病事件探测技术. 大数据, 2016, 2(2): 2016022.

9 蔡皎洁, 张玉峰. 基于语义挖掘的食源性疾病安全预警系统构建. 情报杂志, 2014, 33(2): 18–22.

10 Guo D, Li J, Cao H, *et al*. A collaborative large spatio-temporal data visual analytics architecture for emergence response. IOP Conference Series: Earth and Environmental Science, 2014, 18(1): 012129.

11 Kim Y. Convolutional neural networks for sentence classification. Conference on Empirical Methods in Natural Language Processing. Doha, Qatar, 2014: 1746–1751.

12 Mikolov T, Chen K, Corrado G, *et al*. Efficient estimation of word representations in vector space. arXiv preprint arXiv: 1301.3781, 2013.

13 Bojanowski P, Grave E, Joulin A, *et al*. Enriching word vectors with subword information. Transactions of the Association for Computational Linguistics, 2017, 5: 135–146. [doi: 10.1162/tacl_a_00051]

14 Chen TQ, Guestrin C. Xgboost: A scalable tree boosting system. Proceedings of the 22nd ACM Sigkdd International Conference on Knowledge Discovery and Data Mining. San Francisco, CA, USA. 2016. 785–794.