人立方3.0系统概要设计

撰写人：曹晨雨

2021-10-10

# 概述

## 开发背景

人立方——台湾问题知识服务系统，面向国家和政府情报部门以及心系祖国统一大业的广大人民群众。

基于公开数据源和领域知识的融合，通过实体识别，关系抽取等技术构建以人为核心的异构关系图谱，同时针对公众人物利用新闻、维基百科、社交网站、官方网站等文本数据提取人物、组织特征摘要，收集和分析关联的主题事件及其因果关系，组成领域情报库。综合上述信息，结合实体和关联检索等功能，构建重点人物关系挖掘系统，并进一步地挖掘图谱网络中的特征。以帮助用户了解台湾问题中所涉及的重点人物、组织的基本情况及其相互之间的关联关系。

## 名词解释

1. 新闻文章：指本系统从若干个新闻网站（已有《中时电子报》，未来加入蓝绿阵营的宣传媒体）上所获取到的新闻报道，一般指报道的正文部分；
2. 关键人物：在政治、社会中具有一定影响力的人物，包括地区行政官员、党派关键人物、媒体人、资本巨头、业界名人等；
3. 社会大众评论：指社交网站的非关键人物用户在关键人物用户官方推文下的回复、留言、转发以及其它明确针对某个关键人物的评论；
4. 事件：社会上发生的具有一定影响力的大事，一般应该包括时间、地点、人物、起因、经过、结果等属性，能够从若干篇新闻报道中提取出；
5. 主题事件：拥有相同或类似关键字、围绕着相同实体、在同一或连续的时间内发生的一系列相关事件的集合；
6. 泛化事件：即若干本质相同的事件的抽象，如“人民币下跌”、“美元下跌”、“欧元下跌”等事件都可泛化为“货币贬值”。

## 系统架构

系统由数据获取层、数据存储中心、数据分析处理层、数据可视化层四大层级组成。数据获取层负责对原始数据的获取，包括新闻和社交网站数据两个部分。数据存储中心负责数据的读取和保存，包括图数据库、关系型数据库、分布式文件系统及其配套的存取接口。数据分析处理层负责文本的处理和信息的挖掘。数据可视化层负责数据的展示和与用户的交互。系统整体架构如图1-1所示。

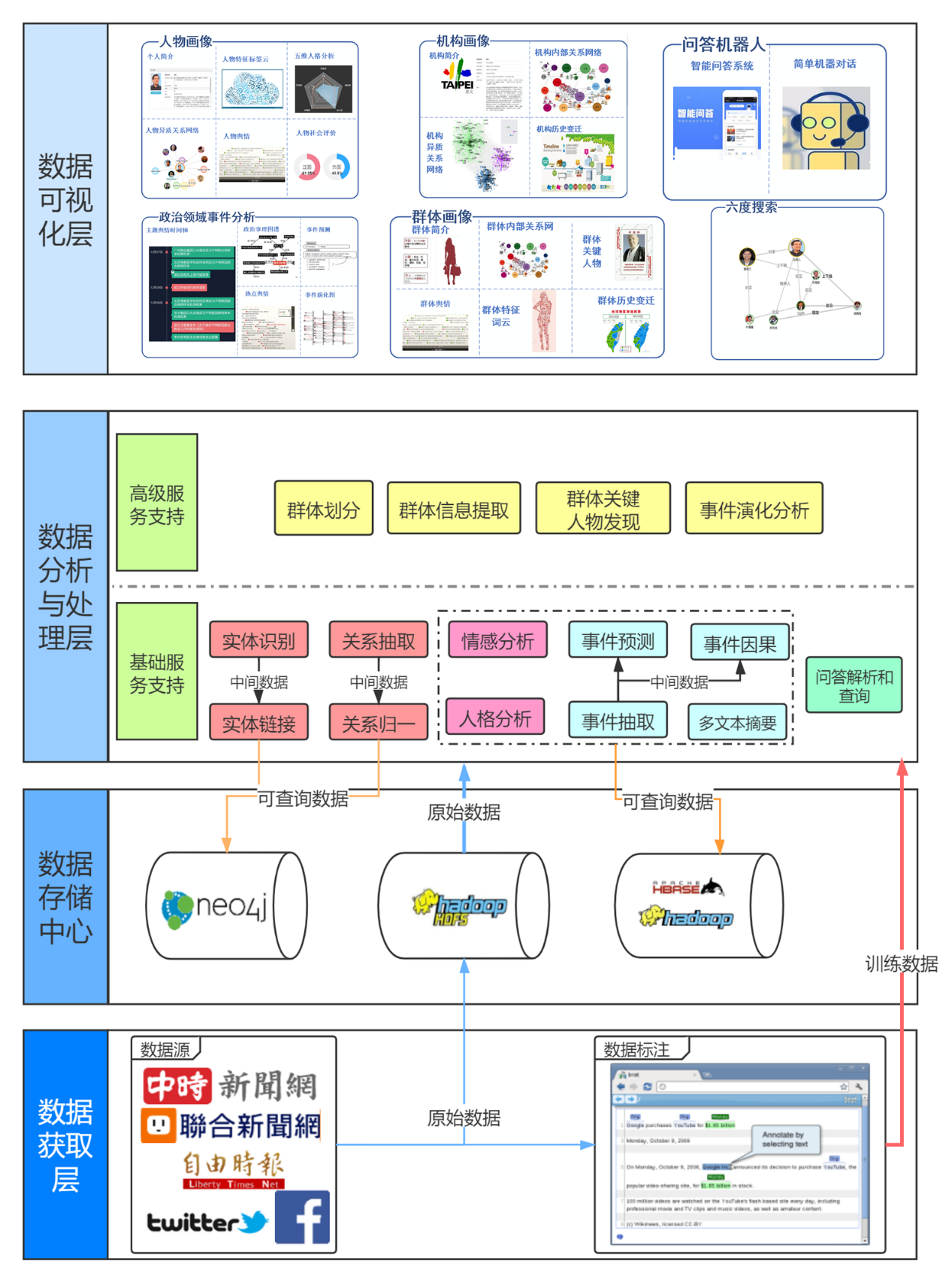


图 1‑1系统架构图

## 系统特色

### 功能特色

本系统共设计大小功能共24项，通过对于市面上同类型系统的调研，本系统所独有的功能共7项，且均系首创，特色功能总结如下：

* 1. 人物画像-人格分析：本系统首次将人格加入了对人物的刻画维度中；
  2. 人物画像-社会评价：本系统首次将人物的社会评价加入了对人物的刻画维度中；
  3. 人物画像-相关事件：本系统充分考虑了政治领域人物画像的特殊性，将人物相关事件（新闻舆情）加入到人物的刻画维度中，并首次设计了事件概述功能；
  4. 机构画像-机构内部关系网络：本系统首次将机构内部人物的关系网络加入了机构画像的刻画维度中；
  5. 事件分析-事件预测：本系统首次设计了基于新闻的政治事件预测功能；
  6. 群体画像-派系划分：本系统首次尝试按照党内派系来对政治人物进行划分；
  7. 群体画像-关键人物：本系统首次设计了群体关键人物发现功能。

### 算法特色

目前概要设计已经考虑实现的核心部分涉及后端算法12项，其中相比于现有算法已经有所创新的有：

1. 句内关系抽取：使用了结合语义标注和图神经网络的开放式关系抽取方法
2. 多文本摘要：使用了基于小规模模型的高效摘要方法
3. 人格分析：使用了基于图神经网络的人格分类方法

与组员研究方向相关，完成时会使用创新算法的有：

1. 实体消歧：基于HITL和知识嵌入的实体消歧方法
2. 关系归一：基于HITL和表示学习的归一化方法
3. 多媒体目标检测
4. 多媒体实体链接和共指
5. 多媒体数据融合
6. 句间关系抽取
7. 事件因果分析
8. 事件预测
9. 问答解析

# 模块设计

## 数据获取层

### 新闻爬虫

利用爬虫程序从新闻网站获取新闻数据，摘取新闻的标题、网址、发布时间、主题、关键词、正文，并以文件的形式保存。新闻文章的正文和关键词部分是后续实体画像和事件抽取的主要数据。目前来自《中时电子报》的数据共有68271篇文章，占用存储空间270MB。如按照相同的规模计算新加入的蓝绿阵营媒体《联合报》和《自由时报》的数据，则应共有20万篇左右的文章，占用存储空间810MB。

### 社交网站爬虫

从社交网站（twitter，facebook）中查找重要人物的账号主页，并爬取人物发布的官方推文和社会大众评论。这些数据将为人物画像中的人格分析和社会评价服务。

## 数据存储中心

使用neo4j图形数据库和Hbase数据库，支持百万级节点和关系的存储管理。

原始数据通过数据中心进行临时存储，在与数据服务的实体信息功能交互后，仅将实体和关系信息存放到neo4j中，其余的实体和关系的附加信息，事件信息等一律存入Hbase，以避免过长的处理流程可能导致的数据丢失，并提高数据库读写效率。在相应数据分析处理完成后，得到的结果先存入数据库，在用户希望查看时再进行读取，除去了因实时处理而造成的响应时延，大大提高了用户的使用体验。

### Neo4j

Neo4j是一个高性能的,NOSQL图形数据库，它将结构化数据存储在网络上而不是表中。它是一个嵌入式的、基于磁盘的、具备完全的事务特性的Java持久化引擎，但是它将结构化数据存储在网络(从数学角度叫做图)上而不是表中。Neo4j也可以被看作是一个高性能的图引擎，该引擎具有成熟数据库的所有特性。程序员工作在一个面向对象的、灵活的网络结构下而不是严格、静态的表中——但是他们可以享受到具备完全的事务特性、企业级的数据库的所有好处。

Neo4j提供了大规模可扩展性，在一台机器上可以处理数十亿节点/关系/属性的图，可以扩展到多台机器并行运行。相对于关系数据库来说，图数据库善于处理大量复杂、互连接、低结构化的数据，这些数据变化迅速，需要频繁的查询——在关系数据库中，这些查询会导致大量的表连接，因此会产生性能上的问题。Neo4j重点解决了拥有大量连接的传统RDBMS在查询时出现的性能衰退问题。通过围绕图进行数据建模，Neo4j会以相同的速度遍历节点与边，其遍历速度与构成图的数据量没有任何关系。

### HBase

HBase是一个分布式的、面向列的开源数据库，该技术来源于 Fay Chang 所撰写的Google论文“Bigtable：一个结构化数据的分布式存储系统”。就像Bigtable利用了Google文件系统（File System）所提供的分布式数据存储一样，HBase在Hadoop之上提供了类似于Bigtable的能力。HBase是Apache的Hadoop项目的子项目。HBase不同于一般的关系数据库，它是一个适合于非结构化数据存储的数据库。另一个不同的是HBase基于列的而不是基于行的模式。

HBase是Apache Hadoop中的一个子项目，Hbase依托于Hadoop的HDFS作为最基本存储基础单元，通过使用hadoop的DFS工具就可以看到这些数据存储文件夹的结构,还可以通过Map/Reduce对HBase进行操作。

HBase不同于一般的关系数据库,它是一个适合于非结构化数据存储的数据库。所谓非结构化数据存储就是说HBase是基于列的而不是基于行的模式，这样方面读写你的大数据内容。

HBase是介于Map Entry(key & value)和DB Row之间的一种数据存储方式。就有点类似于现在流行的Memcache，但不仅仅是简单的一个key对应一个value，你很可能需要存储多个属性的数据结构，但没有传统数据库表中那么多的关联关系，这就是所谓的松散数据。简单来说，你在HBase中的表创建的可以看做是一张很大的表，而这个表的属性可以根据需求去动态增加，在HBase中没有表与表之间关联查询。你只需要告诉你的数据存储到Hbase的哪个column families 就可以了，不需要指定它的具体类型：char, varchar, int, tinyint, text等等。但是你需要注意HBase中不包含事务此类的功能。Apache HBase 和Google Bigtable有非常相似的地方，一个数据行拥有一个可选择的键和任意数量的列。表是疏松的存储的，因此用户可以给行定义各种不同的列，对于这样的功能在大项目中非常实用，可以简化设计和升级的成本。

由于我们项目的源数据包含各种各样的数据，并且都是非结构化的数据，所以HBase很适用于我们系统的源数据存储**。**

## 数据分析处理

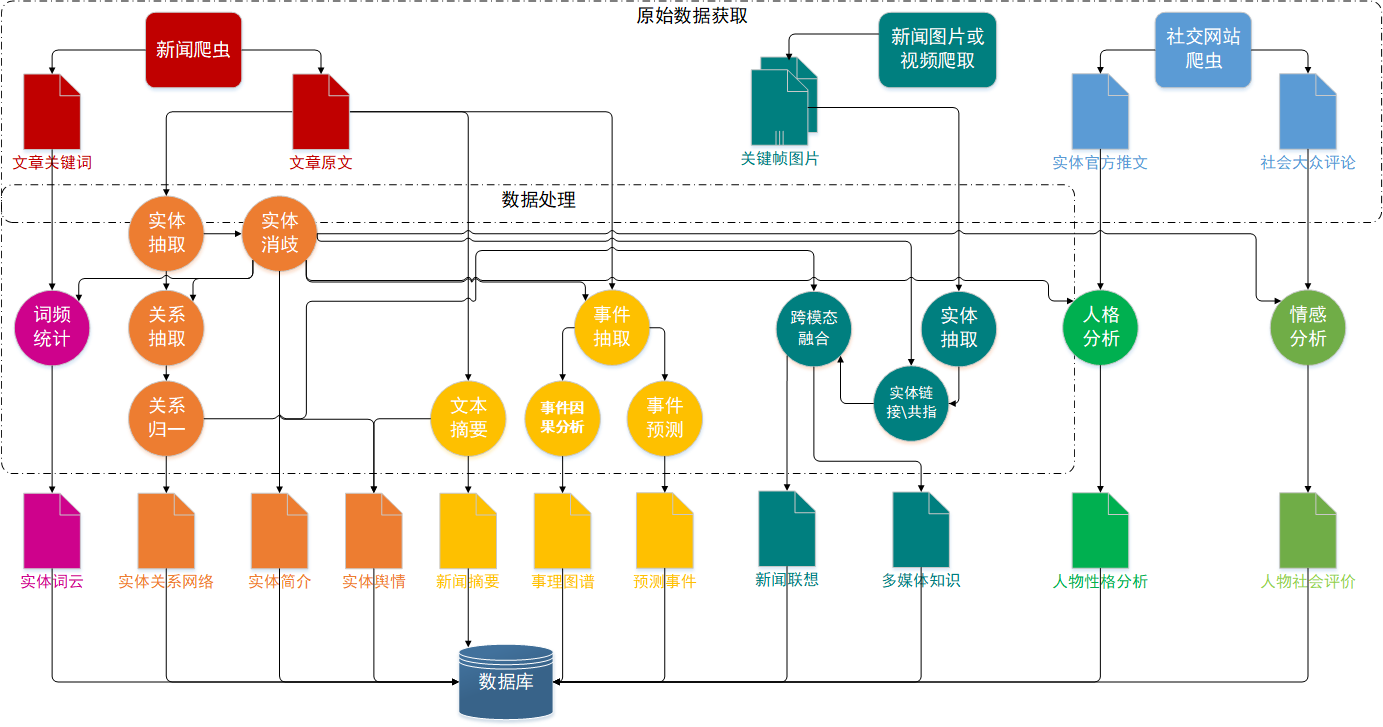


图 2‑1数据分析处理流程图（基础服务支持）

本层的处理流程如上图所示，目前的设计中只考虑了基础服务的支持，群体画像和部分事件分析的功能实现难度较高，故优先考虑了系统的核心算法以及与组员研究方向相关的算法实现。

### 实体抽取

通过基于神经网络的实体抽取算法，以序列标注的形式标记出新闻正文中出现的人物、组织、地点实体，及其所处的句子，以备进一步的处理。

### 实体消歧

使用实体抽取所得到的实体名，在wiki的http接口中进行查询，获取若干个wiki中候选实体的部分属性和简介，通过基于神经网络的实体链接算法，将这些候选实体的名称、属性、简介的语言学特征与待消歧实体的名称、所在句子、所在文章的语言学特征进行匹配，从候选实体中找出待消歧实体所指代的实体。

为适应实际应用的需要，实体消歧模型借用了HITL（Human-in-the-loop annotation）的思路，将数据标注和模型训练两个步骤结合了起来。当一个待消歧的实体名输入模型时，模型首先从wiki的查询接口中获取候选实体，经过与待消歧实体的相似度排序后输出候选实体列表。之后由人工从中选择出正确的实体，此时模型将从人类的选择中进一步进行学习，优化排序策略，如此迭代反复，直到训练完成。预测时则直接将排序后得到的相似度最高的候选实体作为结果输出。训练和预测的流程如下图所示：

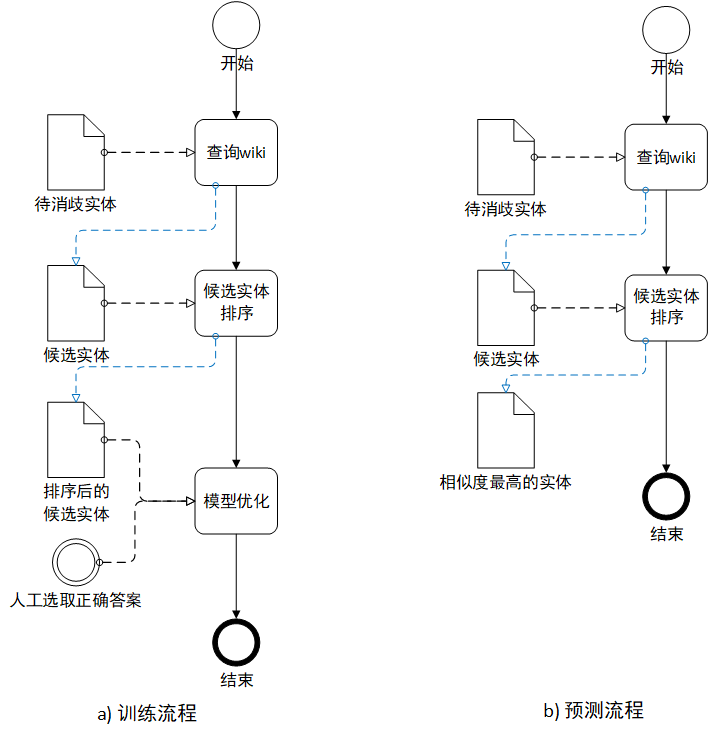


图 2‑2实体消歧训练预测流程图

模型本身也可以提前使用一定量的标准数据进行初步训练，不过在实际的生产实践中，基于机器学习的算法模型往往需要部分实际数据的训练才能够更加适应使用场景，为此需要人工标注一定量的数据。相比于传统的实体消歧模型，这种HITL的方法借用了少样本学习和强化学习的思路，一方面，通过人类手把手的训练，模型可以快速地适应应用场景；另一方面，模型在迭代过程中准确率逐步提高，也让人工标注的工作量随着模型的优化而减少。这种训练方式对于样本不足的领域具有很高的适配性，能够大大减少生产过程中的时间和人力成本。

之后将所指代的实体的属性和简介以及wikipedia的实体编号组合为基础实体对象，将实体对象保存在neo4j数据库中，该实体与所在的文章之间也建立关联，此关联须保存至数据库中。同时，在实体抽取阶段所得到的实体原句中标记实体的wiki编号，以备关系抽取算法的使用，文章中的所有实体对象组合形成文章实体列表，以备其它算法的使用。

### 关系抽取

关系抽取分为句内关系抽取和句间关系抽取两部分算法。

**句内关系：**以实体抽取和实体消歧得到的实体和实体原句为输入，筛选出含有至少两个实体的句子，并将含有超过两个实体的句子中的实体两两组合，最终得到若干个仅标记了两个实体的句子，使用基于语法标注和神经网络的开放式关系抽取算法进行句内开放式实体关系抽取，使用句中的动词、名词、代词、形容词等词语表示实体之间所具有的关系，并与两个实体的wiki编号一起组成关系三元组对象。

此处使用的关系抽取算法采用表示学习的方式，从大量的语料中学习了语法依存关系的表示，同时在语法树上使用了GCN模型，克服了近年来基于神经网络的开放式关系抽取算法无法充分使用语法信息的问题。

开放式关系抽取所得到的关系由于采用的是句子中原有的词语，相比于传统的封闭式关系抽取有着可读性强、鲁棒性强的特点。但其表示上却具有不一致性，即同一对实体间的不同关系词表述的是同一个关系，为此需要对关系进行归一化处理，将表达相同关系的关系词进行表述上的统一，减少冗余的连边。

**句间关系：**以实体抽取和实体消歧得到的实体和文章全文为输入，使用文章级关系分类算法在文章范围内进行跨句子关系抽取，判断不在同一个句子中共现的实体对之间是否也能够通过文章推断出关系，提取出的关系与两个实体的wiki编号一起组成关系三元组对象。

三元组对象主要包括两个实体的wiki编号、名称和关系的名称。句间关系得到的三元组对象本身已经进行了归一化，故将直接被存放在neo4j数据库中，句内关系则需进行进一步得关系归一化。

### 关系归一化

关系的归一化同样借鉴了HITL的思想，算法的核心为关系词的相似度计算模型，我们将本系统的关系归一化操作分为两个阶段，其流程如下图所示：

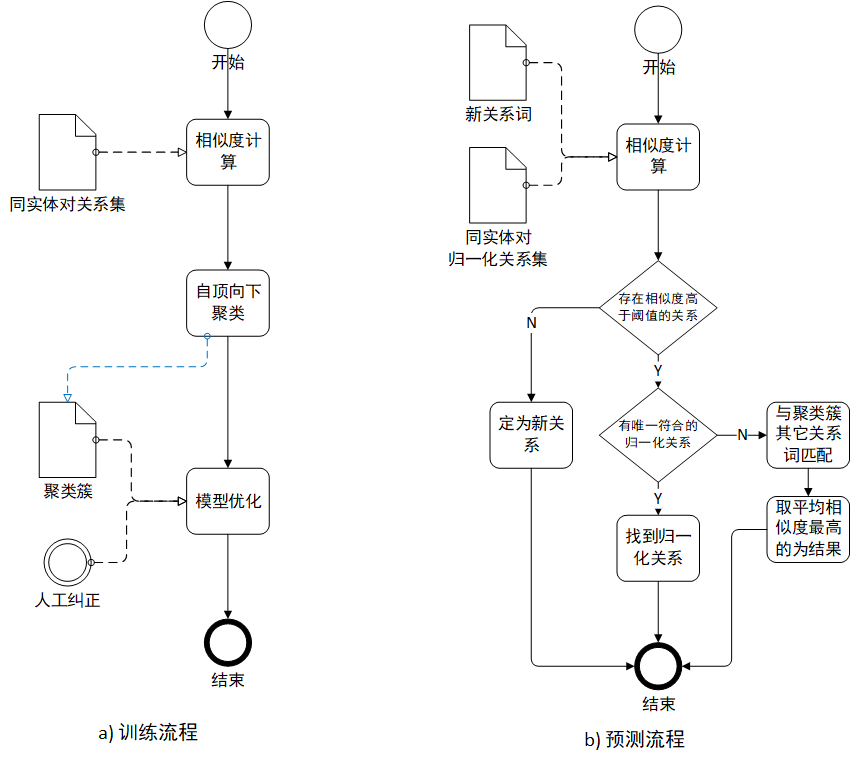


图 2‑3关系归一化训练预测流程图

第一个阶段为训练阶段，在本阶段，系统会将该领域一定规模的历史数据进行开放式关系抽取的处理，所得到的<实体, 关系, 实体>三元组存入文件中备用。当所有的历史数据处理完成后，对于同一个实体对之间的关系，模型会根据每个关系间的相似度进行自上而下的层次聚类，并由人工对聚类错误的点进行重新分配，模型则从人工修改后的结果中进一步优化关系的相似度算法，如此迭代进行，直到训练结束，其余的数据均以聚类的方式进行划分，将每个聚类簇的中心点作为归一化的表述，此时整个聚类簇的归一化关系即为该中心点。

第二个阶段则为预测阶段，与实体链接问题类型，当基础数据的处理完成，系统投入使用并更新数据时，需要将新挖掘出的关系词链接到原有的归一化关系或者新的关系。具体来说，从文章中提取的新关系词将首先与该实体对间已有的归一化关系（聚类簇中心点关系）进行相似度计算：若存在唯一的一个归一化关系与新的关系词相似度大于某个阈值，将新的关系词归类于该归一化关系；若出现超过一个归一化关系与新的关系词相似度大于阈值，则分别计算每个归一化关系所属的聚类簇中其它关系与新关系词的相似度的平均值，取平均值最大者为新关系词的归一化关系；若相似度与所有归一化关系均低于某个阈值，则定为新型关系，新关系词本身就是的归一化关系。

在上述流程中得到的归一化关系将与实体对组成三元组，存储在neo4j数据库中，其它关系与归一化关系将被存储在Hbase数据库中。

### 多媒体目标检测

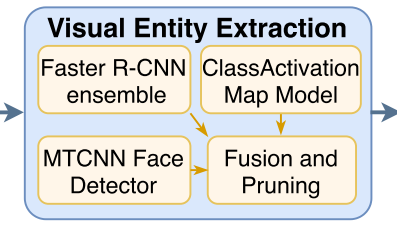


图 2‑4视觉目标检测模块图

视觉上主要需要检测的实体有人体、物体、人脸。

**人体和物体：**采用Faster R-CNN预训练模型抽取图片或视频关键帧中的object，主要思路是先通过启发式方法或RPN网络生成一系列稀疏的候选框，接着对候选框进行分类与回归，综合效率以及检测效果，采用R-CNN系列实例分割与检测算法。

**人脸：**人脸检测是一种在应用场景广泛的算法，可以识别数字图像中的人脸。人脸检测有时也代指人类在视觉场景中定位人脸的过程。在人脸检测中常用且目前比较前沿的算法就是MTCNN（Multi-task Cascaded Convolutional Networks）

**ClassActivationMap(CAM) Model：**生成类激活热力图，辅助特定情境的实体和事件抽取。

文本上的实体抽取可以依赖2.3.1模块进行。

### 多媒体实体链接和共指

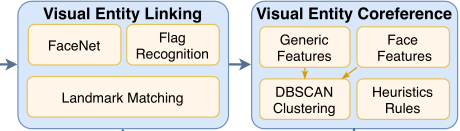


图 2‑5视觉实体链接和共指模块图

视觉上，运用特定应用场景下的分类标签，训练模型专门用于几类标签的视觉实体识别，可通过将开放式图片搜索结果保存到知识库中，来进行识别分类器的单独训练。对于提取到的特征，可以利用DBSCAN Clustering的思想，采用启发式规则聚类进行实体共指。

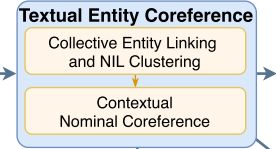


图 2‑6文本实体链接和共指模块图

文本方面，对可链接到知识库中已存在实体的相关信息，添加共指信息。对于无法链接到知识库的相关信息，可以采用NIL Clustering的思路，以启发式规则形成NIL集群，每个集群指向同一个实体。

### 多媒体数据融合

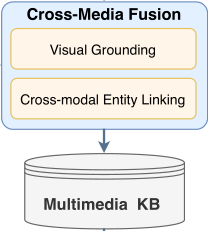


图 2‑7 跨模态融合模块图

为了充分利用多种模态的互补性，可以将多种模态的数据库组合成连贯的多媒体知识库。对于文本中提取的每个实体，将其和整个句子输入到ELMo或BERT等文本特征提取模型中，提取其相关信息及上下文特征，与相关图像特征进行比较，得出相关性得分和热力图，给定阈值以获得足够相关的图像并分配给已知的最重叠的视觉实体或新建一个视觉实体，使用一个类似NIL集群来链接匹配的文本和视觉实体。

### 事件发现

通过对于在同一时间收集到的所有新闻文章进行基于聚类的事件抽取，得到大量事件及其所发生的大致时间。将多篇新闻文章与其所蕴含的事件之间进行关联。与某个事件关联的若干新闻中出现次数最多的若干个实体（即与这些新闻关联最多的若干个实体），将被认为是事件的关键实体，与事件之间建立关联。通过对事件的时间、地点、人物、起因、经过、结果等属性的分析，将其与数据库中已经有的主题事件进行关联，或者与已有的事件组合成为新的主题事件。事件的属性、与实体的关联、所属的主题事件都将被存入Hbase中。

### 事件因果分析

从包含事件的新闻数据中，使用基于规则的方法对事件之间的因果关系进行抽取，得到大量的前置和后置事件对。对于这些事件对进行泛化，除去事件中的特殊属性，抽象出其中的共性，得到大量泛化前置和后置事件集合。对于泛化后的因果事件进行统计，按照事件对出现的频率构建事理图谱，并将泛化后的事件和因果关系存入neo4j数据库中。

### 事件预测

事件预测采用封闭式的事件预测方法，主要研究社会事件在政治领域的影响。当系统获取到一个社会事件时，使用模型预测该事件的发生对于若干个政治经济相关的后置事件的影响。目前的设计中这些事件包括：政府支持率提高/降低，事件人物社会评价提高/降低，社会稳定性提高/降低，民众生活满意度提高/降低。事件预测也将作为事件属性的一部分存入Hbase中。

### 多文本摘要

在得到事件和新闻文章的关联之后，首先将同一事件的相关文章的前部分（总结部分，一般是文章的第一段）结合在一起，组成一篇文章。然后对该文章进行分句、分词处理。将分好词的数据作为模型的输入，最终模型输出关于该事件的摘要。事件与其事件摘要和文章一起组合成事件对象，事件对象需要保存至数据库。

**算法创新：**多文本摘要采用预训练语言模型从大量的语料库中学习到通用的知识。自然语言处理任务在预训练语言模型下都取得了更好的效果。目前的预训练语言模型的趋向于更大、更深、更多参数，但是这对硬件性能，尤其是显存大小有着更高的要求，同时也导致了算法存在鲁棒性差的问题，这使得这些学术成果难以被应用于生产实践。本算法采用预训练语言模型ALBERT。ALBERT采用参数共享等策略，模型的参数量仅为BERT的十分之一，却取得更好的效果。最近对于Transformer-block类型的预训练模型的研究表明：每个隐含层都蕴含丰富的语言，如词序、语法、语义知识。所以算法采用融合层根据所有句子表示计算出ALBERT每一层的权重，融合ALBERT每一层隐含的知识作为最后输出的补充。最终挑选出作为摘要的句子。

### 人格分析

根据实体消歧产生的所有文章的实体列表，经去重后对每个人物实体，手动搜集其社交网站的账号主页，并爬取账号下发布的官方动态，以文件的形式进行保存。再从官方动态中分析并获取人物的大五人格分值，其中，大五人格包括：开放性、尽责性、外倾性、宜人性和神经质性，其通过五种特质涵盖了人格描述的所有方面。人格分析的五维分值将存入关系型数据库的人物实体表中。

针对获取到的每一条动态数据，将属于同一用户的所有动态进行合并，且动态之间使用<sep>分隔符标记。首先对原始合并后的文本，进行数据清洗，使用相关预处理工具（如NLTK）等去除非英文字符，网址链接字符串，用户名字符，表情符等，进而还原缩略词如we’ll →we will等以便后续统计特征信息。然后针对清洗后的文本统计相关特征，用于算法模块的分析。

**算法创新：**在预测人格特质分数此项任务上，现有的研究大多数是根据用户文本抽取其词汇特征，根据词频（如LIWC）等特点，来回归预测五人格分值，但此类方法忽略了用户说话的语义等信息，预测准确度有所偏差。为了弥补上述缺陷，出现了相当一部分研究基于深度学习的方法，运用RNN、CNN等网络捕获文本的语义信息等，再者则结合了LIWC等心理词典特征提升了预测的准确度，但是以往这些方法仅仅是学习文本表层蕴含的语义或词汇特征，其是以传统的文本处理角度去解决该任务，而人格分析不仅仅与浅层的词汇语义等相关联，还与个体发文的惯用的语言表达风格等有着紧密联系，这些也是推断个体人格的重要线索，所以人立方系统中使用的人格分析算法巧妙地结合了语言风格表达等领域知识，将其引入到人格预测这一任务上来，极大地增强了算法的预测能力，也具有很好的可解释性。该算法首先借助美国著名的社会心理学家James W. Pennebaker所著的《语言风格的秘密》一书，人工抽取了词语与风格之间的关系，构建一个词语到风格的层级图，再而根据获取到的数据集中每一个用户统计的特征值（+1或-1），构建与已有心理领域知识图的关联，用户通过这些特征间接与知识图连接，从而形成一个用户-词-语言风格的层次图，采用层次图注意力机制学习用户的表征，具体来说，先将与用户间接关联的语言风格层图输入到GAT网络中，聚集相对重要的语言风格特征，更新语言风格层节点特征，再根据语言风格层与词层节点的关联情况，也通过一个GAT网络更新语言风格-词子图的节点特征，使得与用户关联的词聚集到深层语言风格特征信息，最后使用用户所关联的词节点特征的池化来表征用户，从而接入到全连接层预测五人格分数。

### 社会评价

与人格分析类似，动态搜集其社交网站的账号主页，爬取主页下的大众评论，此外也可以根据关键词、话题等方式搜索其它的大众评论，大众评论以文件的形式保存。再对评论进行情感分析，以获得人物的社会评价，社会评价可以有多种维度。人物的社会评价分值将存入关系型数据库的人物实体表中。

## 数据可视化层

### 知识图谱问答

知识图谱问答属于可视化层的用户交互部分。

输入问题后，根据开放式关系抽取得到的关系划分问题类别，以确定支持问答命令类型。之后根据问题的类型结合实体识别来确定目标实体类型或查询关键词，最终将转化为知识图谱数据库查询来获得结果。

# 数据库设计（暂定）

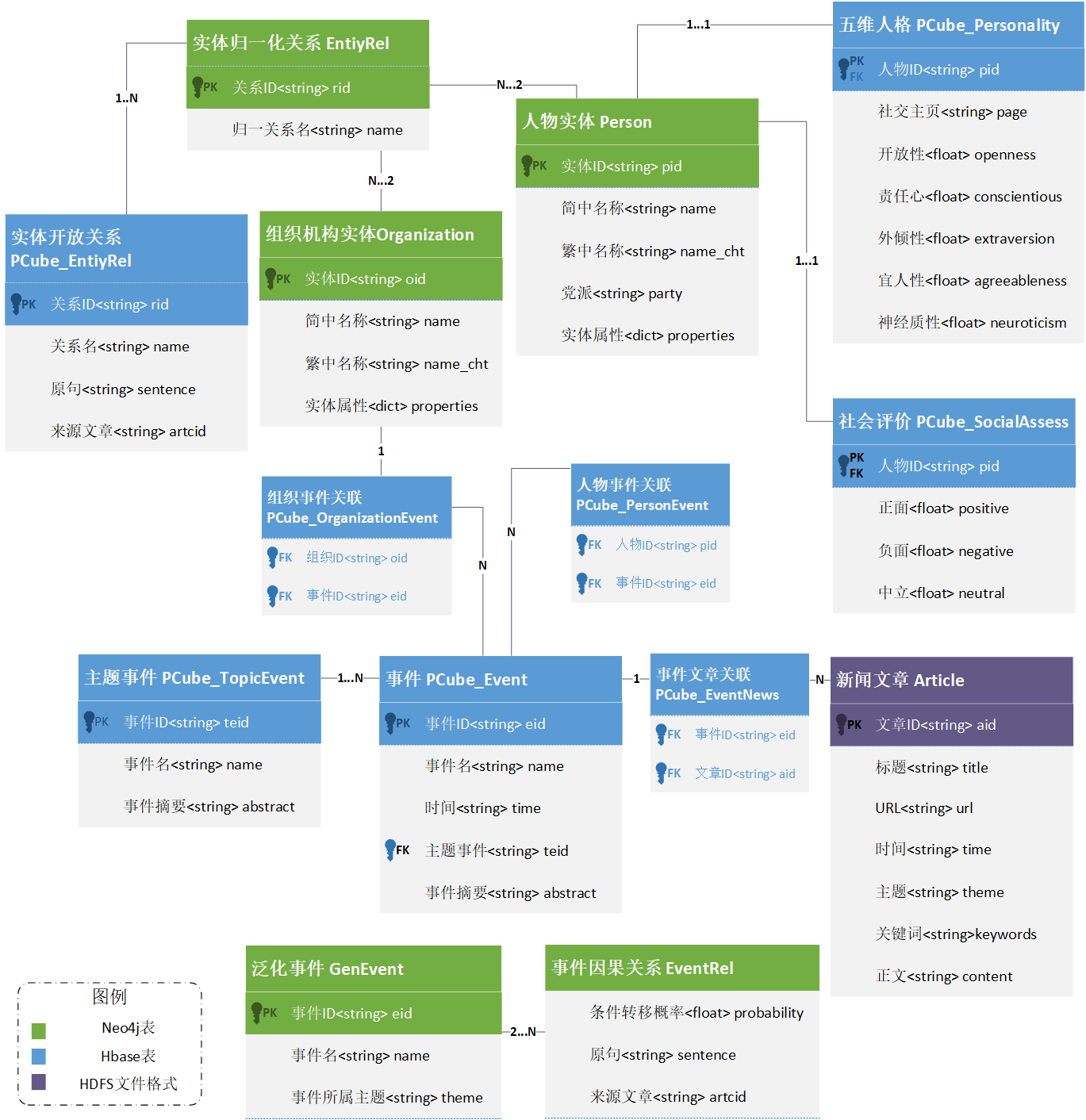


图 3‑1数据库设计图

# 接口设计（暂定）

## 原始数据获取接口

### 新闻爬虫接口

启动爬虫过程，获取新闻数据。

param date: 爬虫目标新闻的最早日期（可选，默认上前一个月）

return: [（news对象）{‘newsId’: 新闻id, ‘title’: 新闻标题, ‘url’: 新闻URL, ‘language’:新闻语言, ‘content’: 新闻内容, ‘topic’: 新闻主题, ‘keyWord’: 新闻关键词}, …]

### 社交网站爬虫接口

启动爬虫过程，获取社交网站数据。

param importantPerson: 爬虫目标重要人物列表

return: [（personPage对象）{‘personId’: 人物id, ‘url’: 主页URL, ‘kind’:网站名称, ‘post’: 帖子内容, ‘comment’: [评论1,评论2, … ] }, …]

## 数据存储接口

### HDFS写入

将爬虫的结果保存为HDFS文件。

param news/personPage: 爬虫结果

return: ‘success’/’error’

### Neo4j写入

param entityAlign: 消歧实体

return: ‘success’/’error’

## 数据服务接口

### 实体抽取接口

实体抽取，序列标注。

param text: 输入文本

return:[‘entity’: {‘entityId’: 实体ID, ‘entityKind’: 实体类型(person/organization/place), ‘entityName’: 实体名, ‘locatedSentece’: 所处句子, ‘entityModal’: 实体模态}, …]

### 实体消歧接口

实体对齐。

param entityName: 输入实体名

return: ‘entityAlign’: {

‘entityId’: 候选实体ID,

‘entityKind’: 候选实体类型(person/organization/place),

‘entityName’: 候选实体名,

‘locatedSentece’: 所处句子

‘locatedArticle’: 所处文章

‘wikiId’: wiki实体编号

}

### 关系抽取接口

param entity: 实体, isSameSentence: true/false 标识是否为句内关系

return: [ ‘entityRelation’: {entity1, relation: 提取出的关系, entity2}, …]

### 关系归一化接口

param [‘entityRelation’…]: 关系三元组列表

return: [ ‘entityRelation’: {entity1, relation: 归一化后关系, entity2}, …]

### 多媒体实体抽取接口

param [picture, …]: 输入图片序列

return:[‘visualEntity’: {‘entityId’: 实体ID, ‘entityKind’: 实体类型(person/organization/place), ‘locatedPicture’: 所处图片及位置}, …]

### 多媒体实体链接和共指接口

**视觉实体链接**

param [‘visualEntity’, …]: 实体抽取输出序列

return: [‘entity’: {‘entityId’: 实体ID, ‘entityKind’: 实体类型(person/organization/place…), ‘entityName’: 实体名, ‘locatedPicture’: 所处图片及位置, ‘entityModal’: 实体模态}, …]

**多媒体实体共指**

param [‘entity’, …]: 实体列表

return: [‘singleModalKnowledge’: {‘knowledgeId’: 知识ID, ‘knowledgeKind’: 知识类型(entity/relation/event), ‘knowledgeName’: 知识名, ‘knowledgeModal’: 知识模态(visual/textual)}, …] 可调用实体消歧及相关接口，生成每个模态的知识

### 多媒体数据融合接口

param [‘singleModalKnowledge’: {‘knowledgeId’: 知识ID, ‘knowledgeKind’: 知识类型(entity/relation/event), ‘knowledgeName’: 知识名, ‘knowledgeModal’: 知识模态(visual/textual)}, …]: 单模态知识列表

return: [ ‘multiMediaKnowledge’: {‘knowledgeId’: 知识ID, ‘knowledgeKind’: 知识类型(entity/relation/event), ‘knowledgeName’: 知识名, ‘knowledgeType’: 粗/细粒度类型}, …]多媒体知识列表

### 事件抽取接口

param text: 文章

return: [ ‘event’:

{‘eventId’: 事件ID,

‘eventName’: 事件名称,

‘eventTime’: 事件发生时间,

‘associatedNews’: [newsId, …],

‘associatedEntity’: [entityId, …]

}, …]

### 事件分析接口

param news 新闻数据

return: [ ‘eventGraph’: {event1, relation, event2}, …]

param news 新闻数据

return: ‘eventPrediction: 事件预测

### 多文本摘要接口

param event 事件

return: {

‘eventId’:事件ID,

‘eventSummary’: 得到的事件摘要,

‘articleId’: {articleTopic, articleURL} 事件包含的文章对象

}

### 人格分析接口

针对清洗后的文本统计相关特征，用于算法模块的分析，具体需统计以下特征：

|  |  |  |  |
| --- | --- | --- | --- |
| 特征 | 特征描述 | 值 | 备注 |
| WC | 字符总数 | -1或1 | 根据用户使用该类词的情况进行赋值，若高于所有用户平均水平则为1，反之则为-1 |
| Pronoun | 代词 |
| I | 第一人称单数 |
| we | 第一人称复数 |
| you | 第二人称 |
| he | 第三人称单数 |
| They | 第三人称复数 |
| Article | 冠词 |
| Prep | 介词 |
| Aux-verb | 助动词 |
| Usual-adverb | 常见副词 |
| Conj | 连词 |
| Negate | 否定词 |
| Verb | 动词 |
| Numeric | 数词 |
| Pos-emotion | 积极情绪词 |
| Neg-emotion | 消极情绪词 |
| Social | 社交有关词汇包括家庭、朋友及两性等 |
| Cognition | 认知过程相关词汇 |
| Insight | 洞察力词汇 |
| Causality | 因果性词汇 |
| diff-verb | 差异动词 |
| Assumption | 猜想性词汇 |
| Certainty | 可能性词汇 |
| Except | 转折词汇 |
| Belonging | 归属性词汇 |
| Achievement | 成就相关词汇 |
| Power | 权力相关词汇 |
| past-tense | 过去时态 |
| now-tense | 现在时态 |
| future-tense | 将来时态 |
| behave-sen | 动作短语 |
| Space | 空间 |
| Time | 时间 |
| big-word | 大词，政治哲学等领域词 |

param : 处理完成后的数据形式

以用户数为单位，一条数据格式如下:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| userId | userName | fmtPost | WC | pronoun | I | … | bigWord |
| str | str | str | int | int | int | int | int |

主要包含用户ID，用户名，处理过的该用户帖子文本，以及上述统计的各类特征等。

数据样例：

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| userId | userName | fmtPost | WC | pronoun | I | … | bigWord |
| u123 | inggwen | post1<sep>post2<sep> | -1 | 1 | 1 | … | -1 |

return:

根据模型得到用户的五人格分值，主要输出数据形式如下：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| userId | userName | opn | con | ext | agr | neu |
| str | str | float | float | float | float | float |

### 社会评价接口

param comment(personPage, …)大众评论

return: {‘evaluation’: 社会评价}

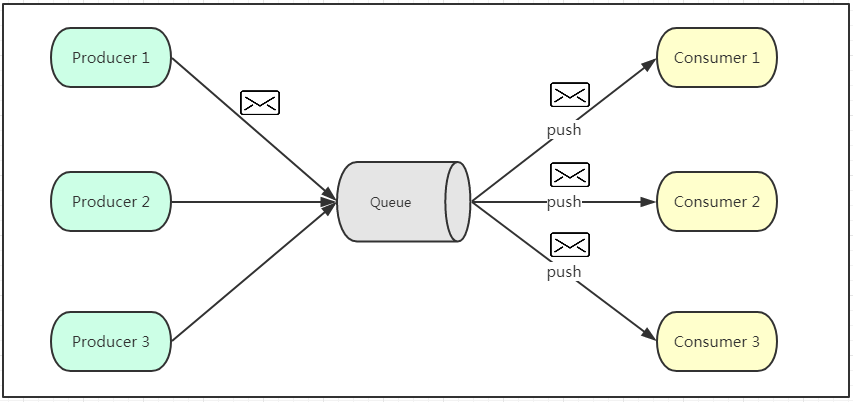
### 知识图谱问答接口

param question问题

return: {‘answerKG’: 知识图谱回答}

# 可提升增量（暂定）

## 数据获取与处理



使用Kafka消息队列进行数据获取与处理，乃至并行化模型的算法测试部分。在Kafka订阅生产者将消息放入消息队列后，队列会将消息推送给订阅过该类消息的消费者。由于是消费者被动接收推送，所以无需感知消息队列是否有待消费的消息。加入Kafka消息队列后，能够有以下的提升：

* 消息系统：解耦和生产者和消费者、缓存消息等。
* 流式处理：比如spark streaming和storm

## 多媒体目标检测及数据融合

在实体链接共指、目标检测等任务上，用多媒体知识进行辅助。视觉上可以用DBSCAN Clustering的思想，采用启发式规则聚类进行实体共指。文本上对可链接到知识库中已存在实体的相关信息，添加共指信息；对于无法链接到知识库的相关信息，可以采用NIL Clustering的思路，以启发式规则形成NIL集群，每个集群指向同一个实体。目标检测方面除了传统的视觉和文本方法，还可以使用ClassActivationMap(CAM)进行辅助

## 群体画像

能够合理进行党派划分与关系提取，目前正在进行数据准备，将由黄彦龙与张振浩进行相关调研与算法实现。

## 情感分析（社会评价）

将根据外网数据进行人物的评论等信息的获取，根据数据分析评论的相关情感及人物的社会评价。

# 人员安排（暂定）

## 算法部分（包含算法模型集成与对应数据获取及处理）

人格分析：朱洋甫

事理图谱：张威

知识问答：张鹤藐

群体画像：张振浩、黄彦龙

情感分析（社会评价）：宋威

实体、关系相关算法：杨卉帆

多媒体相关算法：宋威、杨卉帆

## 后端部分（主要为数据通道，保证并发量与鲁棒性）

开发人员：张振浩、黄彦龙、曹晨雨（晏成昊review代码）

## 前端部分（主要为数据的进一步处理与可视化展示）

开发人员：杨卉帆、宋威（张鹤藐）

## 其他（主要为新技术支持与底层维护）

负责人员：曹晨雨、晏成昊