一. Extract_entity

使用 extract_entity.py 脚本将给定文件中的各种实体进行抽取，其中 data 文件夹用于存放已给的文本，而 label 文件夹用于存放各种实体所对应的标签，而 dict 文件夹用于保存已处理好的抽取出来的实体。

其中用正则表达式来匹配形如“安徽省/LOC”的字符串，分别提取出 value：“安徽省”和其对应的标签 label：“LOC”：

最终保存在 dict 目录下的相应 txt 文件中，其中每一行代表对应的每一则新闻的标签，格式如下：

福建 LOC 浙江 LOC 贵州 LOC 重庆 LOC

二. Build_graph

使用 build_graph.py 脚本根据在 dict 文件夹的数据来建立相应的 neo4j 图数据库，下面是定义的 schema：

Node:

节点名	节点数	属性数
新闻	936	15
日期	374	1
地点	2311	1

Vertice：

关系名	关系 name	关系数
News_begin_days	开始时间	1063
News_end_days	结束时间	1036

News_locs	发生地点	8220

三. Display

在命令行中输入 neo4j console，然后在浏览器中打开 http://localhost:7474

下面可以使用图数据库的 Cypher 语言来对 Knowledge Graph 进行查询和可视化

三. Impovement

还可以在以下方面进行改进：

1. 使用更大的数据集来构建知识图谱
2. 使用预训练模型来根据节点语义对节点进行去重，如该例中的“7月 1 日”和“7.1”
3. 根据 neo4j 查询语句来实现下游任务，通过实现 question parser，question classifier 和 answer searcher 等部件，来实现一个暴雨洪涝的 QA 系统