

# OSpider v3.0.0 用户手册

---

## 1 软件说明

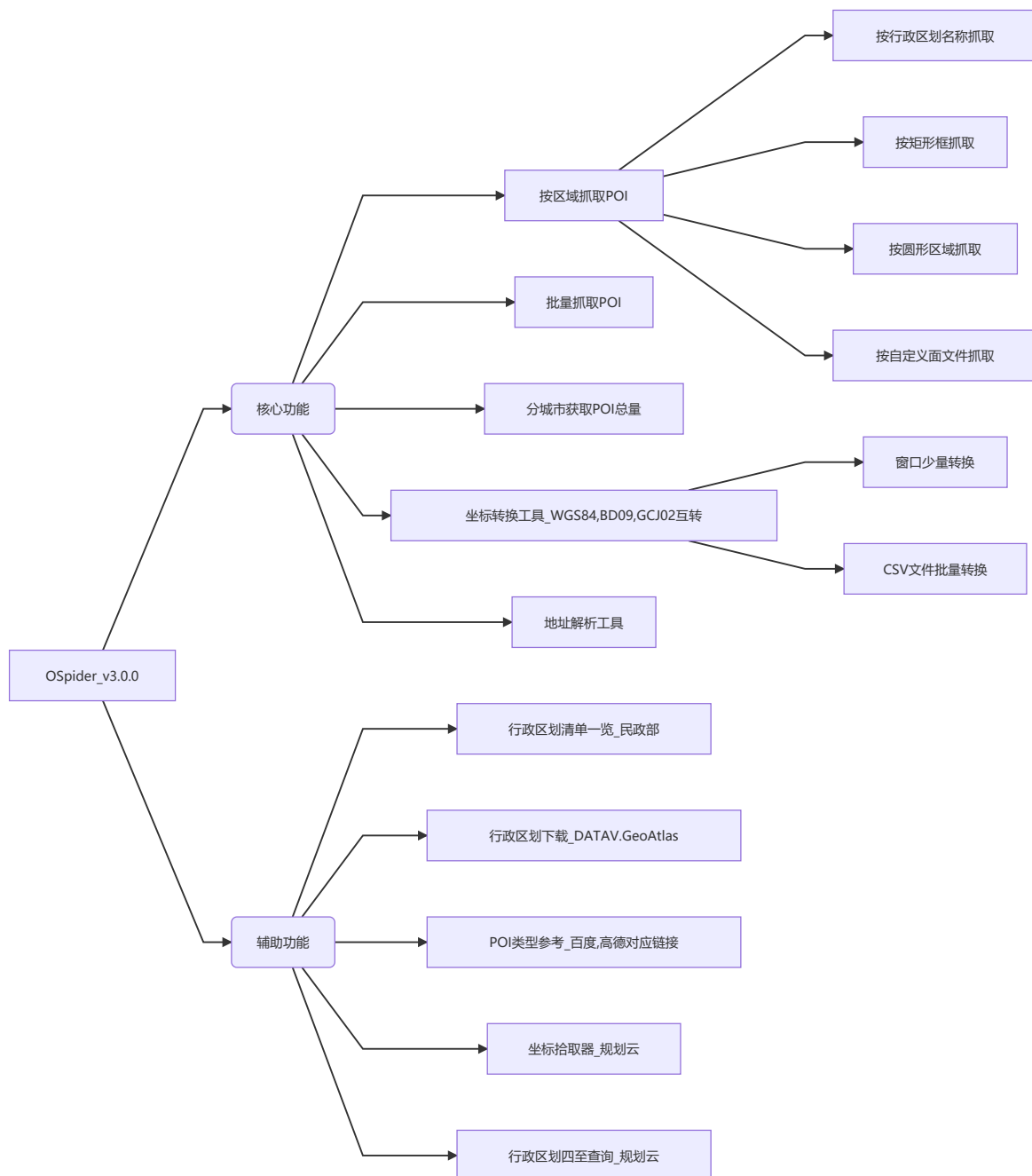
---

### 1.1 总述

OSpider是GPL v3.0协议下的开源桌面软件及python库，致力于提供便捷的矢量地理数据获取和预处理体验。项目主页为：<https://skytruine.github.io/OSpider/>

### 1.2 当前版本功能 (v3.0.0)

当前版本的核心功能为**按行政区划名称、矩形框、圆形区和自定义面文件**四种方式**抓取POI**(暂仅支持百度POI,高德POI将再下一次更新中加入)，支持通过csv批处理文件批量执行POI抓取任务,且提供了分城市获取POI总量的实用工具。OSpider v3.0.0也集成了**WGS84/BD09/GCJ02坐标互转工具**与**地址解析工具**。其他相关功能参见下图：



## 1.3 升级说明

### 版本变迁：v3.0.0<-v2.1.0

- 配置了key池和多线程
- 修改了POI抓取算法
- 支持多模式抓取POI及批处理
- 集成了包括地址解析在内的多个独立工具
- 重构GUI，进一步提升代码质量与用户体验
- 重构并解耦了各功能模块

总的来说从v1到v2是从命令行到GUI的飞跃，从v2到v3是**功能丰富度软件稳定性及代码质量的飞跃**。希望OSpider能持续健康发展。

## 2 使用教程

### 2.1 文件结构说明

名称	修改日期	类型
Demo	2020/9/1 16:18	文件夹
help.pdf	2020/8/20 17:36	PDF Document
icon.ico	2018/7/22 15:28	ICO 文件
OSpider_v3.0.0.exe	2020/8/31 23:54	应用程序
property.ini	2020/9/1 15:54	配置设置

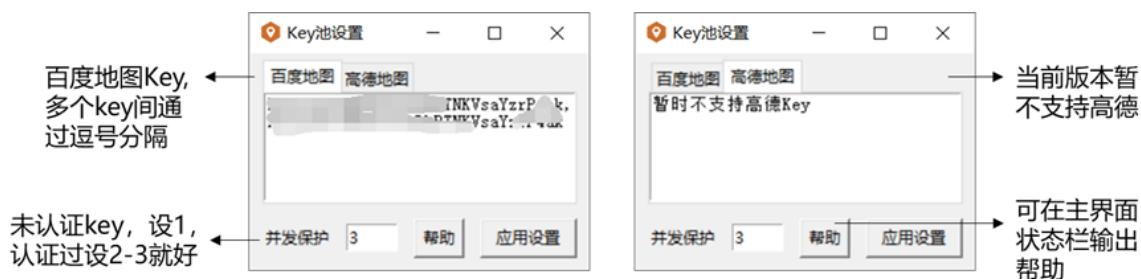
- **OSpider.exe**为OSpider的主程序，双击后弹出OSpider应用主界面。如果杀毒软件报警，请忽略，OSpider很安全。
- **help.pdf**为当前版本的用户手册，帮助用户掌握OSpider操作技巧，对软件使用过程中的常见问题给出解决方案。
- **icon.ico**为OSpider界面的图标，不可删除，删除将导致程序启动异常。
- **property.ini**为OSpider配置文件，内涵与GUI界面关联的各项参数，不可删除，删除将导致程序失去参数及界面大小记忆功能。
- **Demo**为文件输入Demo保存文件夹，也是输出结果的默认保存文件夹，建议保留。

## 2.2 快速操作引导

1. 菜单栏设置Key池后即可正常运行
2. 主界面支持四种方式抓POI，其余工具请在工具菜单中选择
3. 点击各界面中的帮助按钮将会在运行状态栏输出工具使用指南

## 2.3 核心功能详解

### 2.3.0 Key池设置（先决条件）



请在首次运行软件时请点击菜单栏中的“Key池”，完成Key池设置，Key的获取详见FAQ。Key与Key间使用英文逗号“,”分隔，请勿在Key池末尾加空格或者回车换行。在未正确设置Key池的情况下，仅有“坐标转换工具”能够正常运行。

由于API存在调用频率（并发QPS）的限制，为了不触发key的并发禁用，需要进行并发保护。OSpider通过设置“单个Key最多能被多少个线程同时占有”进行并发保护。并发保护数具体设置为多少应当根据网络状况不同而调整，当网速非常快的时候，并发保护超过3就极有可能超并发，对于未经过认证的key而言，并发保护数大于1就存在并发风险。在未触发Key的并发禁用的前提下，并发保护数越大，程序运行越快。

### 2.3.1 按区域抓取POI

OSpider v3.0.0

帮助 工具 关于 Key池

查询关键词

POI名称 酒吧

POI类型

行政区 矩形框 圆形区 自定义

四种抓取模式

POI类型可不填

行政区划名称可简写

区域名称 武汉

区域清单

一般设为4足够

初始网格 4

四分调值 100

未认证单Key设为1, 认证过的可以设为3或更大

线程数 3

数据源 百度地图

输出文件 按区域抓取POI\_result.csv

浏览

帮助

执行

运行状态

设置输出文件为: Demo/按区域抓取POI\_result.csv

状态栏输出帮助

点击执行即开始按区域抓取POI

运行状态会在状态栏持续显示直到抓取完毕

行政区 矩形框 圆形区 自定义

矩形定义

118.351915,29.192178

辅助查询

行政区 矩形框 圆形区 自定义

圆形定义

0248,31.2,36276,3000

辅助查询

行政区 矩形框 圆形区 自定义

文件路径

Demo/上海.json

浏览

支持shp/json/csv/txt

圆心经度,圆心纬度(BD09坐标),半径(m)

支持面Shapefile,GeoJSON,KML

特别注意总评分(overall\_rating)中的-1表示无评分

uid	name	address	province	city	area	tag	telephone	overall_rating	wgs84_lng	wgs84_lat	bd09_lng	bd09_lat	gcj02_lng	gcj02_lat
0 e96e649ce	龙鑫会所	宣城市旌德县	安徽省	宣城市	旌德县	美食;酒吧		-1	118.5444	30.29846	118.5557	30.30216	118.5493	30.29586
1 65bcc7b5f	小镇CLUB	安徽省宣城市	安徽省	宣城市	旌德县	美食;酒吧		-1	118.5406	30.29193	118.552	30.29559	118.5455	30.28933
2 5c4077f11	乌托邦	宣城市绩溪县	安徽省	宣城市	绩溪县	美食;其他	1.51E+10	4.5	118.663	30.07893	118.6743	30.08262	118.6679	30.07634
3 006c72afa	BAILEDI纯城墙路与1	安徽省宣城市	安徽省	宣城市	绩溪县	美食;酒吧		-1	118.5865	30.06761	118.5978	30.07092	118.5913	30.06494

CSV/TXT

Shapefile

结果含WGS84/BD09/GCJ02三套坐标, GeoJSON或Shapefile输出坐标系自动设为WGS84

GeoJSON

```
{
  "type": "FeatureCollection",
  "crs": { "type": "name", "properties": { "name": "urn:ogc:def:crs:OGC:1.3:CRS84" } },
  "features": [
    { "type": "Feature", "properties": { "uid": "cc0849a0ca01525194031964", "name": "judy's", "address": "上海市静安区铜仁路331号-1(近南阳路)", "province": "上海市", "city": "上海市", "area": "静安区", "tag": "美食;酒吧", "telephone": "(021)62893715", "overall_rating": "4.4", "wgs84_lng": 121.44487996340466, "wgs84_lat": 31.22925452138368, "bd09_lng": 121.455909, "bd09_lat": 31.233695, "gcj02_lng": 121.44947231287624, "gcj02_lat": 31.227369035491968 }, "geometry": { "type": "Point", "coordinates": [ 121.444879963404659, 31.22925452138368 ] } },
  ]
}
```

**功能:** 按行政区划名称/矩形框/圆形区/自定义面文件抓取某数据源(百度/高德/谷歌)特定关键词和类型的POI数据并保存为指定类型文件(shp/json/csv/txt) -当前仅支持百度地图为数据源

**输入参数**

参数 #####	说明
POI名称	检索关键字。支持多个关键字并集检索，不同关键字间以\$符号分隔，最多支持10个关键字检索。如:"银行\$酒店"。如果需要按POI分类进行检索，请将分类通过"POI名称"参数进行设置，如"POI名称=咖啡厅"，此时POI类型建议留空。
POI类型	检索分类偏好，可为空，多个分类以","分隔，如果需要严格按分类检索，请留空并设置POI名称为分类名。典型的POI类型使用场景为特定类型下按名称检索，如抓取名称中包括"鑫源"的便利店（POI名称=鑫源，POI类型=便利店）
区划名称	省（不建议）、市、区名称（支持模糊匹配，如输入武汉，代码将自动解析为武汉市）。当要抓区县级行政区时，建议采用城市名+区县名的方式（因为区县级行政区存在重名），如"武汉市洪山区"，行政区名称可通过右侧清单按钮进行查看。
矩形定义	左下经度,左下纬度,右上经度,右上纬度。注意要使用BD09坐标，且分隔符一律为","，可通过右侧辅助按钮进行辅助获取。
圆形定义	圆心经度,圆心纬度,半径。注意要使用BD09坐标，且分隔符一律为","，可通过右侧辅助按钮进行辅助获取。
文件路径	自定义抓取区域面文件的路径，支持shp,json,kml等常用格式,文件坐标系是什么都行，程序内部会自动根据需要进行转换。
初始网格	抓取时首先将抓取区域的外接矩形划分为n*n个切片，n即为“初始网格”。 <b>默认值“4”能满足绝大多数抓取需求</b> ，如果需要在主界面抓多市或省及以上范围，建议设置为最大值“20”。如果设置为大于20的数，在实际执行时将强制重置为20。
四分阈值	当切片返回的POI量大于四分阈值m的时候，将对当前切片进一步四分。对于 <b>百度地图而言设为“100”即可获得极高的POI查全率</b> ，对于高德地图而言设为“850”即可获得极高的POI查全率。
线程数	多线程抓取中启用的线程数，如果线程数>key池大小x并发保护数(在key池中设置)，实际执行时的线程数将被消减至“key池大小x并发保护数”。实际可用线程数越大抓取速度越快。
数据源	抓取POI所使用的数据源，当前仅支持百度地图。
输出文件	结果文件的保存路径，支持输出为Shapefile,TXT,CSV,GeoJSON

### 抓取结果属性说明

列名	说明
uid	唯一标识符
name	POI具体名称
address	POI地址
province	POI所属省份
city	POI所属城市
area	POI所属区县
tag	POI标签（类型）
telephone	POI电话，可能为空
overall_rating	POI总体评分，-1表无，5最高
wgs84_lng,lat	WGS84经纬度
bd09_lng,lat	BD09经纬度
gcj02_lat	GCJ02经纬度
geometry	几何属性

### 2.3.2 POI抓取批处理

批量抓取POI

批处理文件: 抓取POI输入\_Demo.csv

帮助 执行 状态 中止

请复制Demo文件进一步编辑，如果中途报错会将剩余任务存为“输入名\_cover.csv”

加载批处理CSV文件

抓取区域，支持文件路径、行政区划名称、矩形定义、圆形定义，其中矩形和圆形定义的分隔符为英文分号“;”

初始网格

四分阈值 线程数

支持相对路径和绝对路径，路径中请勿出现不存在的文件夹，输出文件无后缀默认生成Shapefile

输出任务剩余量 立即中止批处理

任务ID 关键词 类型

id	query	tag	region	grid_num	threshold	thread_num	outFilePath
1	KTV		https://geo.datav.aliyun.com/areas_v2/bounc	4	100		2 福州市_KTV.csv
2	高中	中学	118.351915;29.192178;120.724682;30.569969	4	100		2 西安外接矩形_高中.shp
3	咖啡厅		121.480248;31.236276;3000	4	100		2 上海市政府周边3km_咖啡厅.json
4	酒吧	酒吧	广州	4	100		2 广州市_酒吧.txt

**功能：**利用批处理文件批量抓取POI。

批处理文件为列名确定的CSV文件，建议用户复制Demo文件后进一步编辑，**批处理文件各列说明**如下：

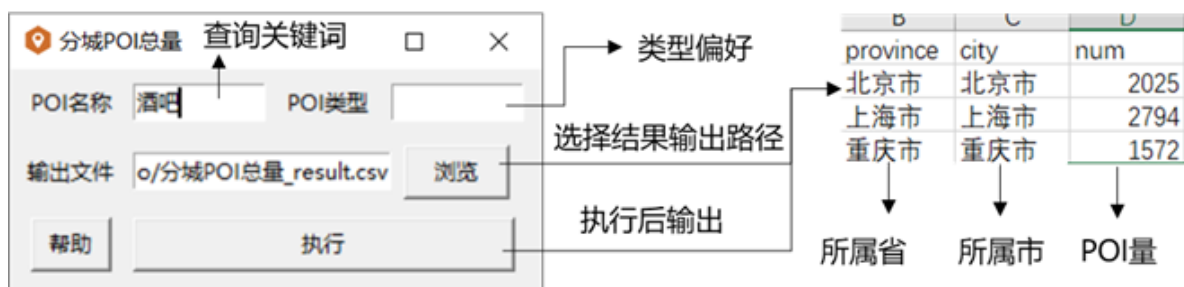
列名	说明
id	任务唯一ID
query	检索关键字。支持多个关键字并集检索，不同关键字间以\$符号分隔，最多支持10个关键字检索。如：“银行\$酒店”。如果需要按POI分类进行检索，请将分类通过query参数进行设置，如“query=咖啡厅”，此时tag建议留空。
tag	检索分类偏好，可为空，如果需要严格按分类检索，请留空并设置query为分类名。典型的tag使用场景为特定类型下按名称检索，如抓取名称中包括“鑫源”的便利店（POI名称=鑫源，POI类型=便利店）
region	抓取区域，抓取区域，支持文件路径、行政区划名称、矩形定义、圆形定义，其中矩形和圆形定义的分隔符为英文分号“;”，定义方式参见“按区域抓取POI”
grid_num	初始网格。抓取时首先将抓取区域的外接矩形划分为n*n个切片，n即为“初始网格”。默认值“4”能满足绝大多数抓取需求。
threshold	四分阈值。当切片返回的POI量大于四分阈值的时候，将对当前切片进一步四分。对于百度地图而言设为“100”即可获得极高的POI查全率，对于高德地图而言设为“850”即可获得极高的POI查全率。（批处理目前仅支持百度地图，故取默认值100即可）
thread_num	多线程抓取中启用的线程数，如果线程数>key池大小x并发保护数(在key池中设置)，实际执行时的线程数将被消减至“key池大小x并发保护数”。实际可用线程数越大抓取速度越快。
outFilePath	结果文件的保存路径，支持.shp/.csv/.txt/.json，无后缀则默认生成.shp

批处理结果文件保存在指定路径中（outFilePath），结果文件的属性与“按区域抓取POI”结果文件一致，本处不再赘述。

### 注意事项

1. 批处理文件region列内部的分隔符为英文分号而不是逗号！！
2. 如果批处理任务中途失败或终止，会将未完成抓取的任务保存为批处理文件目录下的“批处理文件名\_cover.csv”，从而实现断点续传。
3. 由于抓取最后一个任务时不会输出任务剩余量，手动设置了“状态”按钮，点击状态按钮可以查看当前批处理队列还有多少任务
4. 如果想中止批处理，点击“中止”按钮即可，程序会立即中止并生成续传文件。

### 2.3.3 分城市获取POI总量



**功能：**用于抓取除港澳台外国内所有地级及地级以上城市的指定POI数量并输出CSV/TXT格式文件。适用于宏观研究(把各城市的POI量作为指标)。

### 输入参数说明



参数 #####	说明
POI名称	检索关键字。支持多个关键字并集检索，不同关键字间以\$符号分隔，最多支持10个关键字检索。如:"银行\$酒店"。如果需要按POI分类进行检索，请将分类通过"POI名称"参数进行设置，如"POI名称=咖啡厅"，此时POI类型建议留空。
[POI类型]	检索分类偏好，可为空，多个分类以","分隔，如果需要严格按分类检索，请留空并设置POI名称为分类名。典型的POI类型使用场景为特定类型下按名称检索，如抓取名称中包括"鑫源"的便利店（POI名称=鑫源，POI类型=便利店）
输出文件	结果文件的保存路径，支持输出为TXT(.txt),CSV(.csv)

**抓取结果属性说明**

列名	说明
provice	省份名
city	城市名
num	该城市特定POI总量

**注意事项**

1. 分城市获取POI总量所获得的值的与百度地图大比例尺下各城市指定POI数量标注是类似的，**该值仅供参考**，与具体抓取某城市POI的实际数量相比有时偏大有时偏小。
2. OSpider尚不提供城市面文件，如有需要请根据城市名自行链接。

**2.3.4 坐标转换工具**

坐标转换工具

输入栏/坐标系

118.351915, 29.192178  
120.724682, 30.569969

输入坐标系

BD09

输出栏/坐标系

118.34024347058313, 29.16873254934533  
120.71400622741629, 30.56601304221543

输出坐标系

WGS84

少量转换

批量转换

输入文件

D:/坐标转换输入Demo.csv

浏览

输出文件

D:/坐标转换\_Result.csv

浏览

帮助

少量转换结果

选自输入坐标CSV

批量转换输出

输入坐标严格命名为lng和lat

id	lng	lat
1	116.265	36.46213
2	116.4497	37.16367
3	116.3256	37.44962
4	116.3171	37.4449
5	116.3037	37.45482

输入坐标

输出坐标

id	bd09_lng	bd09_lat	wgs84_lng	wgs84_lat
1	116.265	36.46213	116.2529	36.45585
2	116.4497	37.16367	116.4373	37.15716
3	116.3256	37.44962	116.3132	37.44298
4	116.3171	37.4449	116.3048	37.43813
5	116.3037	37.45482	116.2915	37.44784

**功能：**少量或基于文件批量进行BD09/GCJ02/WGS84坐标互转。

**输入参数说明**

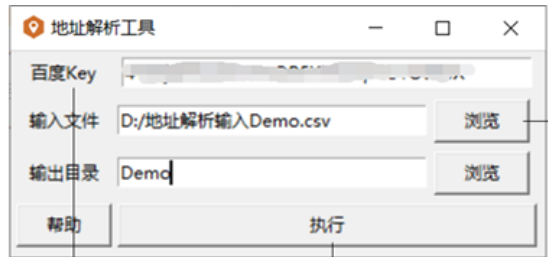


参数 #####	说明
输入栏	少量转换输入经度,纬度,每行表示一个点
输出栏	少量转换输出经度,纬度,每行表示一个点
左侧下拉框	输入栏/或输入文件的坐标系。 WGS84:wgs84坐标系, GPS及谷歌地图所使用的坐标系, 无加密, 常用 GCJ02:gcj02坐标系, 又称火星坐标系, 由国测局对WGS84坐标进行加密得到, 高德地图使用该坐标系 BD09:bd09坐标系, 百度地图使用的坐标系, 对GCJ02二次加密获得
右侧下拉框	输出栏坐标系/及输出文件希望包含的坐标系
输入文件	批量转换的输入csv文件, <b>包括id,lng,lat三个必须列</b> , 其中lng指输入经度, lat指输入纬度。输入文件中允许包含其他用户自定义列, 这些列将在输出文件中得到保留
输出文件	批量转换结果保存文件 (仅支持txt,csv), <b>必定包含id, 及原始和转换后的经纬度5列</b> , 如果输入文件中存在用户自定义列, 这些列将在结果中得到保留

### 注意事项

1. 输入的csv文件要求要转换的坐标必须放置在lng和lat两列中, 请检查你是否将要转换坐标的列名改为了对应值

## 2.3.5 地址解析工具



选择待解析文件

id	city	address
1	武汉市	武汉大学资源与环境科学学院
2	武汉市	珞狮路63号
3	武汉市	珞喻路与珞狮路交叉口
4	襄阳市	诸葛亮广场水上世界市
5	茂名市	文昌居委会
6		华中师范大学
7		石牌岭社区居民委员会

该工具需单独设置百度地图Key

执行并将结果输出至指定目录中的“输入文件名+地址解析结果.csv”

id	city	address	status	wgs84_lng	wgs84_lat	bd09_lng	bd09_lat	precise	confidence	comprehensive	level
1	武汉市	武汉大学	0	114.3609749	30.5398	114.3729	30.5398	1	70	100	教育
2	武汉市	珞狮路63	0	114.3506112	30.53334	114.3625	30.53334	1	80	100	门址
3	武汉市	珞喻路与珞	0	114.3478767	30.52891	114.3598	30.52891	0	80	100	交叉路口
4	襄阳市	诸葛亮广	0	112.1277677	32.0587	112.1404	32.0587	0	50	100	旅游景点
5	茂名市	文昌居委	0	110.9496624	22.37964	110.9608	22.37964	1	70	100	政府机构
6		华中师范	0	114.3552674	30.51999	114.3672	30.51999	0	70	0	教育
7		石牌岭社	0	114.3358186	30.52326	114.3478	30.52326	0	50	0	商圈

应该使用的结果坐标      精度信息, 用于判断是否应该采用或进一步人工打点

**功能:** 根据文件输入, 将地址解析为坐标

### 输入参数说明

参数 #####	说明
百度Key	百度地图的Key，独立于Key池，需单独设置
输入文件	必须严格遵循对应Demo文件格式， <b>必须包含id，city，address三列</b> 。其中city指地址解析城市偏好，address指待解析地址。
输出目录	这里选目录就好，输出文件自动保存为该目录下的“输入文件名+地址解析结果.csv”

**待解析文件说明**

列名	说明
id	地址唯一标识符
city	地址所在的城市名。用于指定上述地址所在的城市，当多个城市都有上述地址时，该参数起到过滤作用，但不限制坐标召回城市。 <b>该参数可为空</b> ，为空时不限制坐标召回城市。
address	待解析的地址。最多支持84个字节。 可以输入两种样式的值，分别是： 1、标准的结构化地址信息，如北京市海淀区上地十街十号【推荐，地址结构越完整，解析精度越高】 2、支持“路与路交叉口”描述方式，如北一环路和阜阳路的交叉路口 第二种方式并不总是有返回结果，只有当地址库中存在该地址描述时才有返回。

**地址解析结果文件列说明**

列名	说明
id	唯一标识符
city	POI具体名称
address	POI地址
status	抓取状态码，0为成功抓取，非0为抓取失败。建议对所有非0结果重新抓取
wgs84_lng,lat	解析结果，地址对应的WGS84坐标
bd09_lng,lat	解析结果，地址对应的BD09坐标
precise	位置的附加信息，是否精确查找。1为精确查找，即准确打点；0为不精确，即模糊打点。
confidence	描述打点绝对精度（即坐标点的误差范围）。 confidence=100，解析误差绝对精度小于20m； confidence≥90，解析误差绝对精度小于50m； confidence≥80，解析误差绝对精度小于100m； confidence≥75，解析误差绝对精度小于200m； confidence≥70，解析误差绝对精度小于300m； confidence≥60，解析误差绝对精度小于500m； confidence≥50，解析误差绝对精度小于1000m； confidence≥40，解析误差绝对精度小于2000m； confidence≥30，解析误差绝对精度小于5000m； confidence≥25，解析误差绝对精度小于8000m； confidence≥20，解析误差绝对精度小于10000m；
comprehension	描述地址理解程度。分值范围0-100，分值越大，服务对地址理解程度越高 <b>（建议以该字段作为解析结果判断标准）</b> ； 当comprehension值为以下值时，对应的准确率如下： c=100，解析误差100m内概率为91%，误差500m内概率为96%； c≥90，解析误差100m内概率为89%，误差500m内概率为96%； c≥80，解析误差100m内概率为88%，误差500m内概率为95%； c≥70，解析误差100m内概率为84%，误差500m内概率为93%； c≥60，解析误差100m内概率为81%，误差500m内概率为91%； c≥50，解析误差100m内概率为79%，误差500m内概率为90%； //解析误差：地理编码服务解析地址得到的坐标位置，与地址对应的真实位置间的距离。
level	能精确理解的地址类型，包含：UNKNOWN、国家、省、城市、区县、乡镇、村庄、道路、地产小区、商务大厦、政府机构、交叉路口、商圈、生活服务、休闲娱乐、餐饮、宾馆、购物、金融、教育、医疗、工业园区、旅游景点、汽车服务、火车站、长途汽车站、桥、停车场/停车区、港口/码头、收费区/收费站、飞机场、机场、收费处/收费站、加油站、绿地、门址

### 注意事项

1. 作者写这个功能偷懒了,依然是单线程单Key，用户有强烈需求才会改
2. 不同的研究和应用对地址解析的经度要求不同，用户完成地址解析后可根据precise和comprehension进行基本的经度判断，考虑是否对部分解析效果好的地址进一步进行手动解析。手动解析可综合使用[百度地图](#)、[高德地图](#)及[坐标拾取器](#)。

## 2.4 常见问题解答 (FAQ)

- **如何获取和设置Key?**

进[百度地图控制台](#)创建APP即可获取对应的Key，建议创建服务器端的APP并获得对应的Key。另外，请及时进行百度地图开发者认证——未认证状态下的Key并发上限和每日额度都太低了，非常容易超并发和抓一两类POI就耗尽了。如果你有条件，建议多收集身边小伙伴的Key，组成Key池——这会使得你抓POI的速度和每日总限额获得极大的增幅。目前抓百度POI、地址解析都是基于百度的Key的。

- **为什么我的Key总是会失效?**

如果不是因为你个人抓取或Key泄露被别人用来抓取了过多的POI从而达到了Key的每日额度上限，则很可能使因为你尚未进行百度地图的开发者认证，从而使得Key的并发上限太低，在抓取过程中多次触达并发上限。**建议将“Key池”选项中的并发保护设置为1，并多放几个Key在里面。**

- **为什么抓POI的时候会卡在“正在提取区域内的POI”?**

核心原因是在当前代码中“提取区域内的POI”这一步非常耗时，目前尚未找到非常好的替代方式，考虑未来版本中予以优化。以“按行政区划名称”抓取“武汉市”的POI为例，如果我们抓的是武汉市的咖啡厅，武汉市外接矩形内会有1040个左右的咖啡厅，进一步把980个武汉市内的咖啡厅提取出来大概会耗时几秒到十几秒的时间；如果我们抓的是“中餐厅”，那么外接矩形中会有34000+个，进一步提取出武汉市内23561个中餐厅大概会耗时3-5分钟；如果我们直接抓“美食”，那么外接矩形中会有约7万个对应POI，进一步提取出武汉市内52984个“美食”可能会耗时5-10分钟（因具体设备而异）。由于提取操作是用过第三方库Geopandas库完成，因此这里没有设置输出，故而一旦提取时间比较长，你可能就会觉得卡主了。**请耐心等待，过一会就提取完了。**如果你希望这个提取操作非常短，那么建议你要抓的行政区的四至来按矩形框抓取——这样的话提取操作就由你自己在ArcGIS或其他软件汇总完成了。在等待提取的时候，你可以喝杯咖啡，或者多开OSpider执行其他任务。

- **如何抓的更快?**

在影响抓取精度（切分模式）的“初始网格”和“四分阈值”（一般采用默认初始网格=4，四分阈值=100就足够应对绝大多数情况了）一定的情况下，提高抓取速度的不二良方无疑是使用大Key池、调高并发保护(不建议设置超过5，一般认证过的key可以设置为2-3)、增加主界面线程数。**实际抓取速度与“实际可用线程数”成正比，实际可用线程数=min(设置线程数, Key池大小 x 并发保护)。**

- **我能否用OSpider抓全省乃至全国的POI?**

**可以。**直接在主界面抓全省或全国单类POI时请将初始网格设置为支持的最大值“20”，虽然可以，但我们不推荐这样做，这会使得抓取时间极长且中间任意环节出现问题将前功尽弃。**我们更推荐的做法是使用“批量爬取POI”工具**，你可以把全国或全省的大任务拆分成一个个城市的子任务，这样可以实现子任务抓取结果的即时保存及断点续传。

- **我能否一次抓取同一区域内的多种POI?**

当然可以，有两种方式一次抓取同一区域内的多种POI。①主界面直接抓取：直接在“POI名称”栏输入用“\$”分隔的多种POI（如“酒吧\$咖啡厅\$KTV”）并保持“POI类型”为空即可，抓取的不同POI将保存在同一文件中，抓取完毕后需要手动分离。这种方式适用于三五种量相对较少（非几千上万）的POI一起抓的情况。②批量爬取POI工具：详见对应工具说明，这种方式是我们更推荐的。

- **我如何按照分类抓取POI?**

将“POI名称”设置为分类名即可，此时建议POI类型取空，如（POI名称=酒吧，POI类型=），可在菜单栏->工具->POI类型参考获取具体的POI类型取值。此外，我们不建议你直接抓取大类POI，这可能会使抓取时间过长。我们推荐你使用“批量抓取POI”工具对大类POI进行分拆。

- **为什么矩形框/圆形区抓取报错？**

最有可能的情况是你**未按照要求定义矩形或圆形**，矩形通过左下经纬度右上经纬进行定义，以英文逗号“,”进行分隔；圆形通过圆心经纬度和半径进行定义，以英文逗号“,”进行分隔。典型报错案例是因为用户输入了分号“;”进行分隔或者在定义末尾有空格或回车换行。

- **为什么我获得的POI数量与百度或高德地图上显示的不一致？**

不论是百度还是高德提供的POI查询API均存在一定的数据滞后及数据抽稀，因此我们几乎不可能获得当前的全量POI，只能通过抓取算法的改进**尽可能逼近“存在一定滞后的全量POI”**。OSpider在当前所有可用的POI抓取代码和软件中几乎具有最高的查全率了，扩充Key后抓取速度也十分OK。（暂限于百度地图数据源，高德的数据源将在未来加入）

- **为什么抓取的POI和我的底图对不上？**

几乎可以肯定是坐标系不一致所导致的。在ArcMap中，请设置数据框的坐标系为WGS84，这能解决投影坐标系与地理坐标系的显示偏差问题；此外需要指出各类地图下载器下载的影像栅格可能有偏，如谷歌有偏和高德实际上是GCJ02坐标系，百度的影像则是BD09坐标系，这种情况下请在加载POI的时候定义坐标系为WGS84，但X,Y栏使用对应的GCJ02坐标或BD09坐标。此外需要注意，“定义投影”和“投影”是两码事，“定义投影”并不会更改实际坐标，“投影”会更改实际坐标——如果你弄来弄去底图和POI怎么都叠不在一起，请想想自己是否曾错把“定义投影”当“投影”来用了。

- **为什么抓取的POI中会有重复值，我该如何处理重复值？**

由于API存在返回部分抓取区域外的POI及同名POI的情况，抓取结果中可能存在重复值。抓取后根据实际情况利用uid和POI名称手动去重即可——Excel删除重复值功能了解一下。

- **为什么POI抓取批处理会报错**

如果不是Key池的原因，那么其他主流原因包括两个，第一文件编码问题：建议直接复制Demo文件夹下的对应样例csv文件并在此基础上进行修改；第二region列定义问题：请格外注意，region列中矩形框和圆形区定义的分隔符是英文分号“;”，这与主界面中使用的逗号分隔符是不同的。

- **为什么分城市获取POI总量工具会报错？**

针对部分POI如“北京烤鸭”，由于数据模板不一致，所以可能报错，由于绝大多数POI能够正常运行因此该Bug尚未得到修复。

- **为什么批量坐标转换工具会报错？**

主流出错原因包括两点，第一文件编码问题：建议直接复制Demo文件夹下的对应样例csv文件并在此基础上进行修改；第二列名问题：**输入的csv文件要求要转换的坐标必须放置在lng和lat两列中，请检查你是否将要转换坐标的列名改为了对应值。**

## 3 关于

### 3.1 开发团队

OSpider项目由小O发起和负责，当前开发团队包括：

- 华盛顿大学HGIS Lab | 小O
- 西南交通大学 | 苟强

### 3.2 加入我们

我们需要对POI、AOI、Land use、路网及其他GIS/规划相关数据获取及预处理有一定了解和实践经验，并希望为开源社区做贡献的开发者小伙伴-Talk is cheap. Show me the code. 如果你认为OSpider的文档编写、Web主页存在不足，且有能力进行改进，我们也非常欢迎你的加入。有加入意向的小伙伴请发送邮件至[ospider\\_org@163.com](mailto:ospider_org@163.com)

### 3.3 Bug报告与意见反馈



如果您在使用OSpider的过程中发现Bug或对OSpider的使用有什么意见或建议，请向[@163.com](https://ospider.org)发送邮件，我们收到邮件后会尽快回复。另外，非常欢迎加入OSpider用户群（QQ）：939504570。

### 3.4 支持我们

[GitHub Star](#)是对我们的最大肯定，而您的赞助支持将为项目的平稳发展保驾护航



生活好 支付宝



\*\*璠

打开支付宝[扫一扫]  
OSpider-支持一下

推荐使用微信支付



舒璠(\*\*璠)

OSpider-支持一下