

## 1. 概述

云知声语义云平台,是一套先进的语义解析,智能问答和智能对话服务的技术接入平台。它以自然语言理解为基础、以问答对话为最终产品形态,面向微信运营服务提供商,以及中小微开发者,提供语义解析,智能问答和智能对话服务接入。

### 1.1. 目的

本文档是以一个简单的例子为线索,介绍如何使用语义云接口进行语义解析应用的开发,使开发者快速掌握使用语义云的基本方法。

文档读者为使用语音 SDK 进行开发的产品设计师、软件工程师。

### 1.2. 范围

本文档定义语义解析的使用说明、体系结构、API接口、开发实例。不包含核心引擎的性能定义,也不包含其它配套或附赠产品的使用说明。

## 2. 开发准备

### 2.1. 获取 APPKEY

对于个人开发者使用语音服务,需要经过我们的授权,请到“<http://sc.hivoice.cn/nlucloud/>”注册成为我们的开发者,申请应用,获取所开发的应用的 appkey、secret。具体步骤请查看“语义云平台使用指南”(链接)

### 2.2. 环境搭建

这里将在Windows上的安装环境的搭建步骤简要介绍如下:

- 一、安装JDK (Java Development Kit), 读者可以从官网下载所需的版本;
- 二、下载语义云平台 SDK(链接)

## 3. 开发实例

### 3.1. 天气服务开发实例

样例程序中 `MainApplet.java` 是一个 applet 的示例程序,开发者可以直观的了解语义理解结果的解析。`Demo.java` 是一个很简洁的示例程序,里面描述了应该怎样调用语义云的过程。

初始化“请求参数对象”，设置当前应用的固有参数 appkey、secret、appver。

```
NluRequestPo requestPo = new NluRequestPo(appkey, secret);  
requestPo.setAppver(appver);
```

设置语义理解过程中需要的参数，此处参数可以根据实际情况进行相应设置，本例出于简单考虑，设定为固定值。

```
// 下面的参数，如果在语义理解过程中不需要这些信息，可以不设置  
// 设置城市信息  
requestPo.setCity(getCity());  
// 设置GPS坐标  
requestPo.setGps(getGps());  
// 设置时间  
requestPo.setTime(getCurrentTime());  
// 设置终端设备的唯一标识，用于标记当前用户  
requestPo.setUdid(getUdid());  
// 设置 history 信息，在对话模型中，history 信息很重要  
requestPo.setHistory(history);
```

填充需要进行语义解析的内容，例如“今天北京的天气怎么样”，如下图：

```
// 查询天气的问句（如：明天上海的天气怎么样？后天北京会下雨吗？周五会下雨吗？）  
requestPo.setText(questionStr);
```



已经设置完毕的请求对象 `requestPo` 作为参数，调用语义云 API，返回语义理解的 JSON 结果

```
// 云知声语义云调用SDK
YzsNluClient sdk = new YzsNluClient();
// 获取语义理解结果（Json 格式的字符串）
String nluResult = sdk.getNluJsonResult(requestPo);
```

根据上一步返回的 Json 结果并参照云平台 API 规范文档（[链接下载](#)），进行语义理解结果解析，获取的 Json 结果的数据结构如下：

```
{
  "rc": 0, "text": "北京明天天气",
  "service": "cn.yunzhisheng.weather", "code": "FORECAST"
  "semantic": {
    "intent": {"focusDate": "2013-10-13", "city": "北京", "cityCode": "101010100"}
  },
  "data": {
    "header": "北京明天的天气是：小雨,12 至 19℃,微风。",
    "result": {
      "weatherDays": [{
        "year": 2013,
        "month": 10,
        "day": 12,
        "dayOfWeek": 7,
        "weather": "多云转阵雨",
        "highestTemperature": 23,
        "lowestTemperature": 14,
        "currentTemperature": 18,
        "wind": "微风",
        "dressIndex": "舒适",
        "dressIndexDesc": "建议着长袖 T 恤、衬衫加单裤等服装。年老体弱和
        宜着针织长袖衬衫、马甲和长裤。",
        "carWashIndex": "不宜"
      }],
      ...
    },
    "cityName": "北京",
    "updateTime": "Oct 12, 2013 11:00:51 AM",
    "focusDateIndex": 1
  }
},
"general": {
  "type": "T",
  "text": "北京明天的天气是：小雨,12 至 19℃,微风。"
}
}
```

从返回的 Json 结果中获取需要的信息。本例为了简单所以仅获取 `general` 下的 `text` 中的信息，如下：

```
JsonParser jsonParser = new JsonParser();
```

```
JsonObject jsonObj = (JsonObject)jsonParser.parse(nluResult);
//获取json中的相应值
String returnStr=jsonObj.getAsJsonObject("general").get("text").getAsString();
```

即可得到提问的回答为：“北京今天的天气是：晴，0 至 10℃，北风 3-4 级转微风”，如下图：



## 3.2. 航班服务开发实例（对话式）

初始化“请求参数对象”，设置当前应用的固有参数 appkey、secret、appver。

```
NluRequestPo requestPo = new NluRequestPo(appkey, secret);
requestPo.setAppver(appver);
```

设置语义理解过程中需要的参数，此处参数可以根据实际情况进行相应设置，本例出于简单考虑，设定为固定值。

```
// 下面的参数，如果在语义理解过程中不需要这些信息，可以不设置
// 设置城市信息
requestPo.setCity(getCity());
// 设置GPS坐标
requestPo.setGps(getGps());
// 设置时间
requestPo.setTime(getCurrentTime());
// 设置终端设备的唯一标识，用于标记当前用户
requestPo.setUdid(getUdid());
```

```
// 设置 history 信息，在对话模型中，history 信息很重要
requestPo.setHistory(history);
```

填充需要进行语义解析的内容，例如“我要查航班”，如下图：

```
// 查询航班的问句（如：明天从北京到上海的航班，下周一下午到上海的航班，我要查航班）
requestPo.setText(questionStr);
```



已经设置完毕的请求对象 `requestPo` 作为参数，调用语义云 API，返回语义理解的 JSON 结果

```
// 云知声语义云调用SDK
YzsNluClient sdk = new YzsNluClient();
// 获取语义理解结果（Json 格式的字符串）
String nluResult = sdk.getNluJsonResult(requestPo);
```

根据上一步返回的 Json 结果并参照云平台 API 规范文档（[链接下载](#)），进行语义理解结果解析，获取的 Json 结果的数据结构如下：

```
{
    "rc": 0,
    "text": "我要查航班",
    "service": "cn.yunzhisheng.flight",
    "general": {
        "type": "T",
        "text": "请问到哪个城市（比如：上海、广州、深圳）?"
    },
    "history": "cn.yunzhisheng.flight"
}
```

从返回的 Json 结果中获取需要的信息。获取 general 下的 text 中的信息，如下：

```
JsonParser jsonParser = new JsonParser();
JsonObject jsonObj = (JsonObject)jsonParser.parse(nluResult);
//获取json中的相应值
String returnStr=jsonObj.getAsJsonObject("general").get("text").getAsString();
```

提问“我要查航班”没有明确目标城市，语义不明确，所以系统将会返回询问式的语句，如上述提问的回答即为：“请问到哪个城市（比如：上海、广州、深圳）？”，如下图：



按照其询问再次输入内容，如：“上海”，见下图：

```
// 查询航班的问句（如：明天从北京到上海的航班，下周一下午到上海的航班，我要查航班）
requestPo.setText(questionStr);
```



根据上一步返回的 Json 结果并参照云平台 API 规范文档（链接下载），进行语义理解结果解析，获取的 Json 结果的数据结构如下：

```
{
  "rc": 0,
  "text": "去上海",
  "service": "cn.yunzhisheng.flight",
  "general": {
    "type": "T",
    "text": "请问哪天出发（比如：后天、下周一、8月15日）?"
  },
  "history": "cn.yunzhisheng.flight"
}
```

从返回的 Json 结果中获取需要的信息。获取 general 下的 text 中的信息，如下：

```
JsonParser jsonParser = new JsonParser();
JsonObject jsonObj = (JsonObject)jsonParser.parse(nluResult);
//获取json中的相应值
String returnStr=jsonObj.getAsJsonObject("general").get("text").getAsString();
```

上一次回答“上海”虽然明确目标城市，但没有明确航班日期，语义不明确，所以系统将会再次返回询问式的语句，如上述回答即为：“请问哪一天出发（比如：后天、下周一、8月15日）？”，如下图：



按照其询问再次输入内容，如：下周一

```
// 查询航班的问句（如：明天从北京到上海的航班，下周一下午到上海的航班，我要查航班）  
requestPo.setText(questionStr);
```



返回的 Json 结果及云平台 API 规范文档（[链接下载](#)）,进行语义理解结果解析，获取的 Json 结果的数据结构如下：



```

{
    "rc": 0,
    "text": "下个星期二下午",
    "service": "cn.yunzhisheng.flight", "code": "FLIGHT_ONEWAY",
    "semantic": {
        "intent": {
            "origin": "CURRENT_CITY", "destination": "上海",
            "departDate": "2013-10-15", "departTime": "12:00:00,18:00:00"
        },
        "normalHeader": "已经为您查询 2013-10-15 当前城市到上海的航班."
    },
    "history": "cn.yunzhisheng.flight"
}

```

获取 semantic 中的信息，根据语义解析结果拼接成 URL（去哪儿网）

```

JsonElement jsonElement = jsonObj.getAsJsonObject("semantic").get(
    "intent");
JsonObject elementMap = jsonElement.getAsJsonObject();
//获取航班参数
String origin = elementMap.get("origin").getString();
//起点城市
if("CURRENT_CITY".equals(origin)){
    origin ="北京";
}
//终点城市
String destination = elementMap.get("destination").getString();
//航班日期
String departDate = elementMap.get("departDate").getString();
//航班时间
String departTime = elementMap.get("departTime").getString();
// 设置去哪儿网航班信息查询URL模板
initUrl = new StringBuffer("http://flight.qunar.com/site/oneway_list.htm");
String searchDepartureAirport = "?searchDepartureAirport=";
String searchArrivalAirport = "&searchArrivalAirport=";
String searchDepartureTime = "&searchDepartureTime=";
String searchArrivalTime = "&searchArrivalTime=";
String nextNdays = "&nextNdays=0&startSearch=true";
String from = "&from=qunarindex";
// 根据实际情况填充URL模板
initUrl.append(searchDepartureAirport)
    .append(URLEncoder.encode(origin, "utf-8"))
    .append(searchArrivalAirport)
    .append(URLEncoder.encode(destination, "utf-8"))
    .append(searchDepartureTime).append(departDate)
    .append(searchArrivalTime).append(departTime)
    .append(nextNdays).append(from);
returnStr = jsonObj.getAsJsonObject("semantic").get("normalHeader")

```

```
.getAsString();
```

即可得到提问的回答为：“已经为您查询 2013-11-25 当前城市到上海的航班，点击按钮【查看结果】，查看具体信息”，如下图：



点击【查看结果】，即可访问拼接的 URL,得到“下周一从北京到上海的航班”信息

去哪儿网换号啦！1010 1234，全新客服电话，直接拨打，不加区号，24小时为您服务！

去哪儿? Qunar.Com 聪明 你的 旅行

登录 注册 | 我的订单

首页 机票 酒店 团购 度假 火车票 攻略 旅图 门票

单程 城市 从 北京 日期 往 2013-11-25 周一

往返 到 上海 返

去哪儿提供的链接

69元住布丁时尚大床房  
www.podinn.com

0元住酒店更送机票  
www.7daysinn.cn

77元住五星级大床房  
www.7daysinn.cn

11-22 周五	11-23 周六	11-24 周日	11-25 周一	11-26 周二	11-27 周三	11-28 周四	价格 日历
¥382	¥199	¥259	¥301	¥207	¥340	¥259	

北京 → 上海 | 单程

搜索 502 家网站，其中 190 家有直飞报价，195 家有联程报价，共 8064 个报价信息，搜索结束

起飞时间 ☐ 上午 ☐ 中午 ☐ 下午 ☐ 晚上

方式 ☐ 直达 ☐ 中转联程

更多筛选条件

临近航线低价推荐 天津-上海 2013-11-25 ¥99 0.9折 查看

航班信息	起降时间	起降机场	准点率/平均延时	机票推荐	最低报价 (不含税费)	
上海航空FM9108 波音757(中)	07:35 09:45	首都机场T2 虹桥机场T2	约100%		¥301 2.7折	<a href="#">订票</a>
上海航空FM9106 波音757(中)	20:30 22:45	首都机场T2 虹桥机场T2	88% 8分钟		¥301 2.7折	<a href="#">订票</a>
东方航空MU5182 空客330(宽体机)	22:00 23:55	首都机场T2 虹桥机场T2	68% 15分钟		¥302 2.7折	<a href="#">订票</a>
东方航空MU5183 空客330(宽体机)	07:25 09:55	首都机场T2 浦东机场T1	约100%		¥302 2.7折	<a href="#">热门 订票</a>