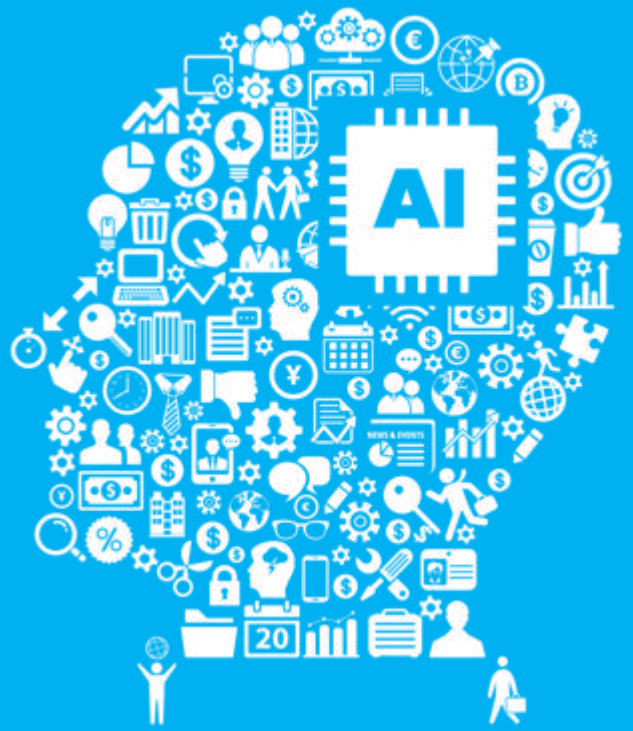
Chatopera
企业聊天机器人

Clause 开源的语义理解系统

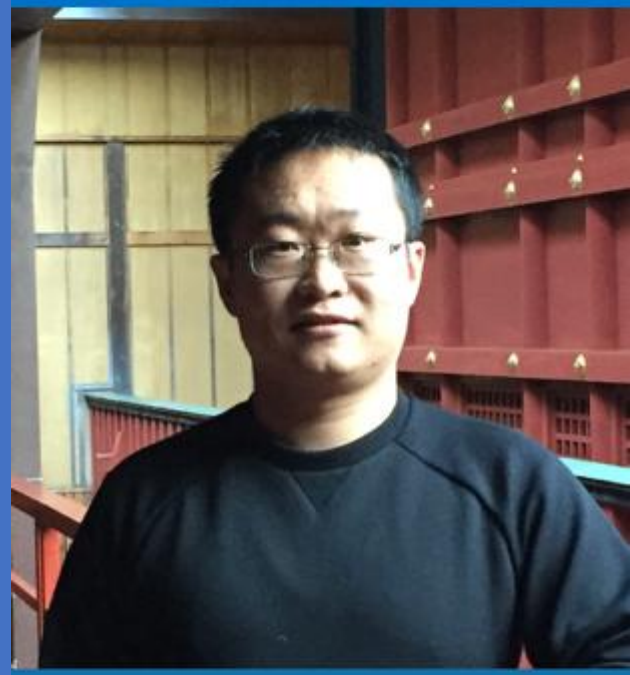
@Bot5 2019-09-27

王海良 Chatopera联合创始人&CEO



自我介绍

王海良，Chatopera联合创始人&CEO，微软人工智能最有价值专家。2011年毕业于北邮，后加入IBM工作四年，先后工作于软件开发实验室和创新中心。从2016年开始工作于创业公司，三角兽AI算法工程师，呤呤英语AI产品负责人，负责智能对话系统研发，2018年出版《智能问答与深度学习》一书。



Microsoft® Most Valuable Professional

奖励分类
AI

首次获奖年份：
2018

MVP 奖励数量：
2

内容大纲

一、概述语义理解服务的背景知识

二、介绍Clause项目的设计与实现

三、开源的Clause

四、基于Clause快速构建一个对话应用

五、总结

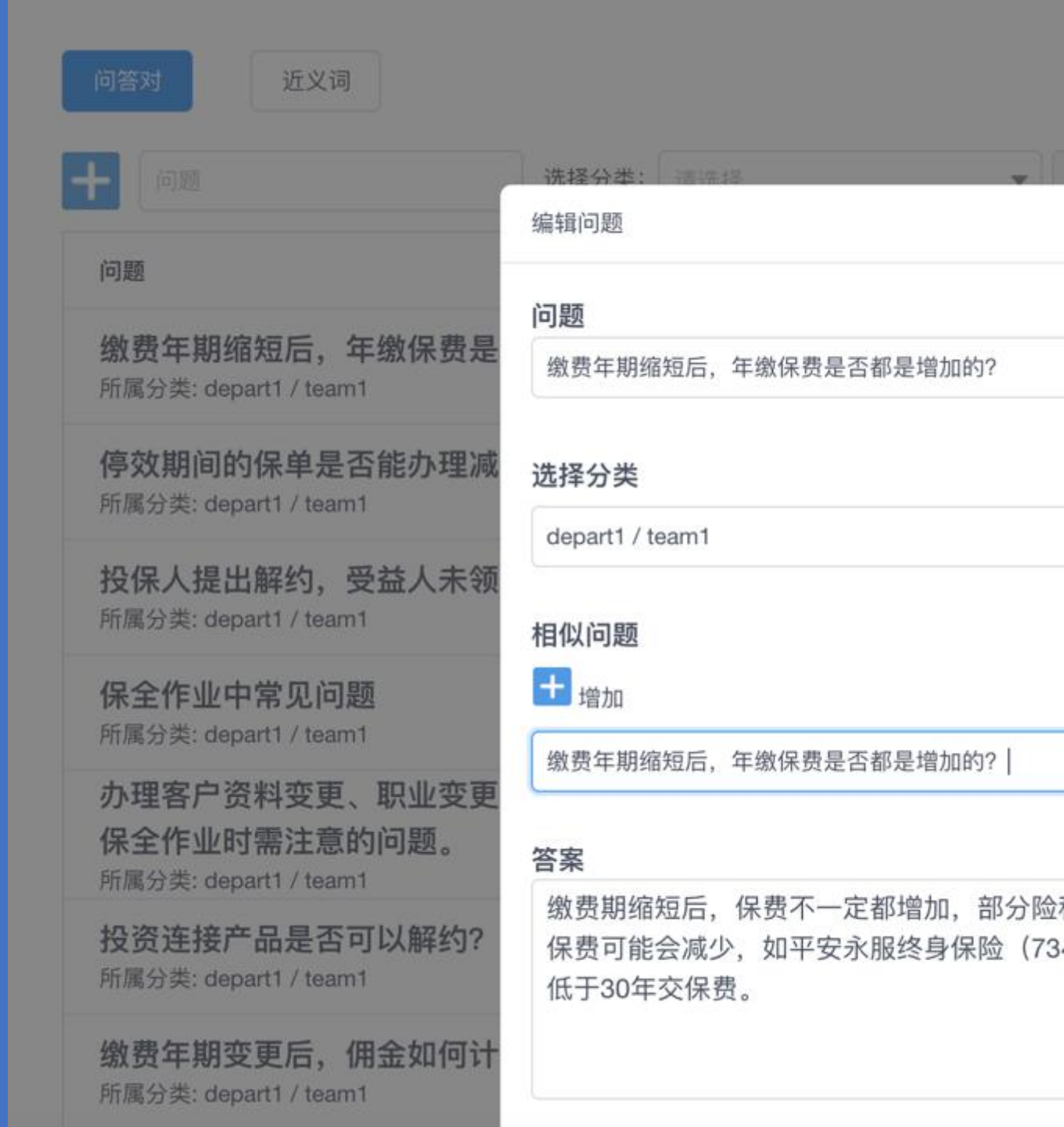
一、背景知识


让机器更懂人

对话类型

基于知识库问答

- 录入问题和答案
- 设计标准问题的相似问题
- 使用搜索和排序，根据相似度返回结果



对话类型

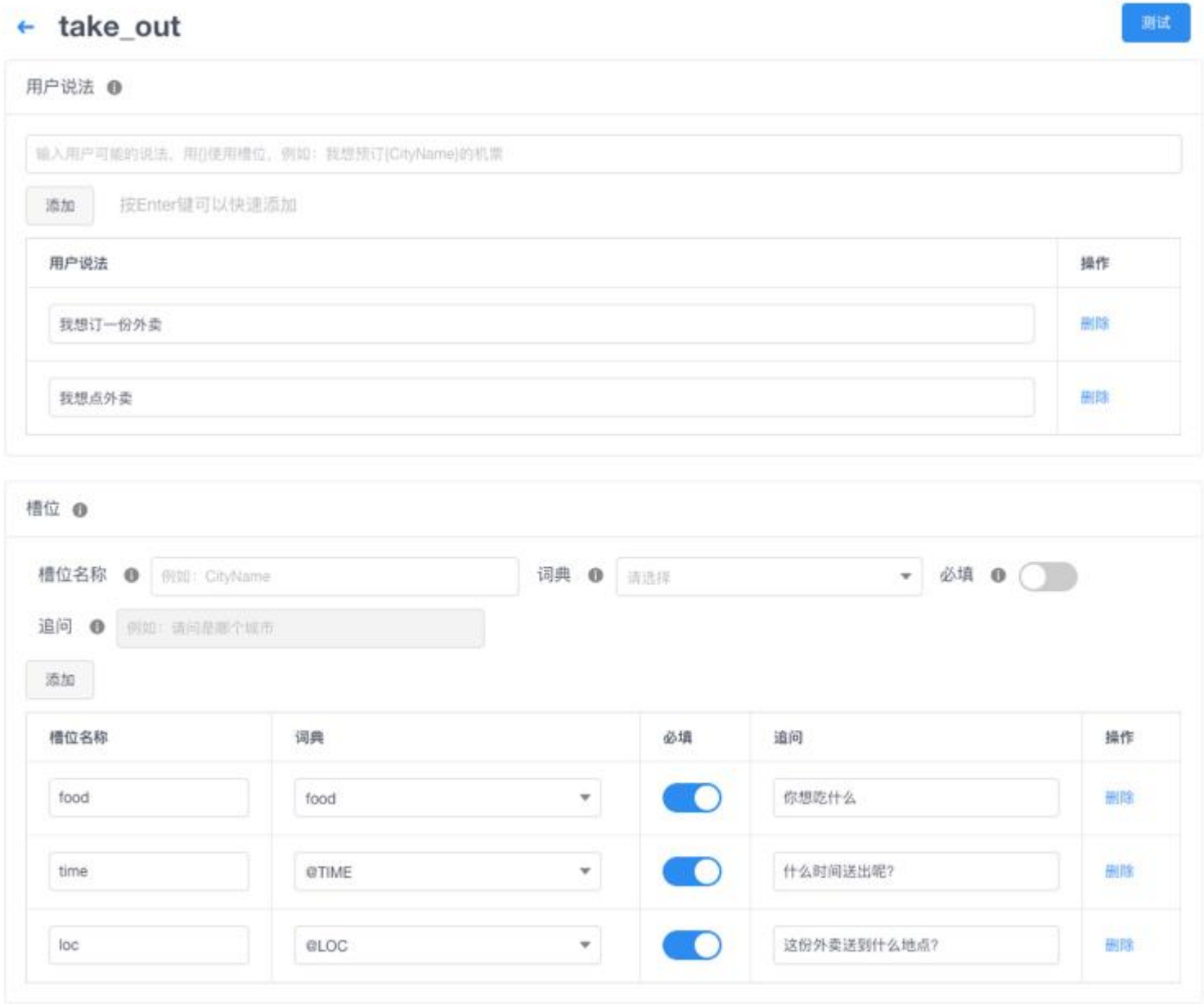
基于规则脚本

- 以一定的语法定义规则
- 常用正则表达式等约束
- 通过钩子追踪槽位

对话类型

基于意图识别

- 使用说法定义意图分类模型
- 利用序列标注识别槽位信息

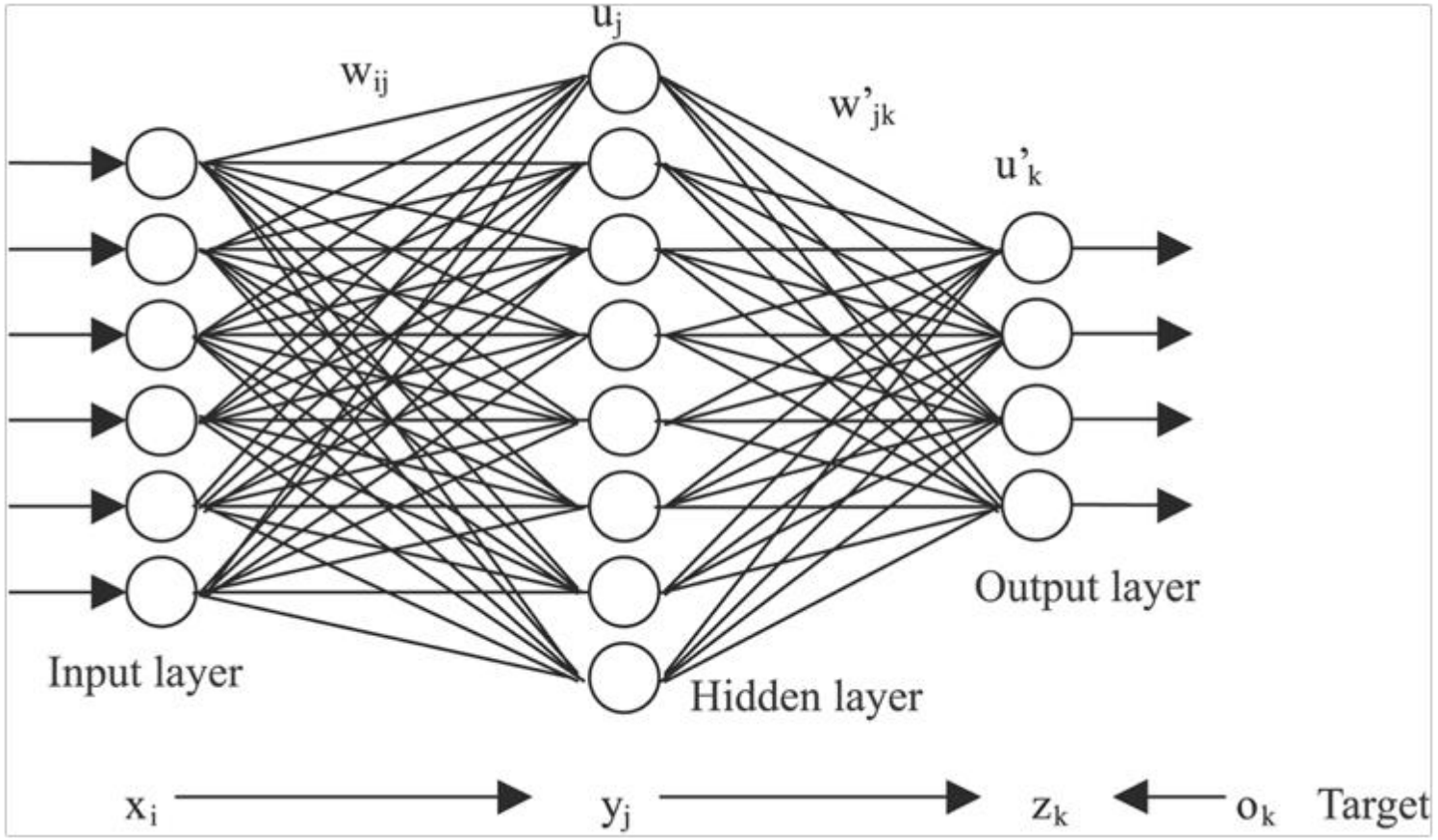


Paradigm shift

统计机器学习应用于文本处理任务

- 分词 / Part of Speech
- 命名实体识别 / NER
- 序列标注 / Sequence Labeling
- ...

人工神经网络



学习路线

入门课程、工具书

[手把手教你打造智能问答模型]

在线课程介绍

第一章 数学基础

第二章 Python编程语言基础

第三章 深度学习初步

第四章 深度学习深化

第五章 智能问答模型实践

配套教材

课程福利

EDU
CSDN学院 IT实战派

直播福利：

1、《从零开始深度学习》-手把手打造智能问答模型

2、原价99元课程，直播福利仅需“59元”

3、输入优惠码：“926”，领取课程优惠券

扫码输入优惠码，立即购买

添加小姐姐入群，领取课件+录播

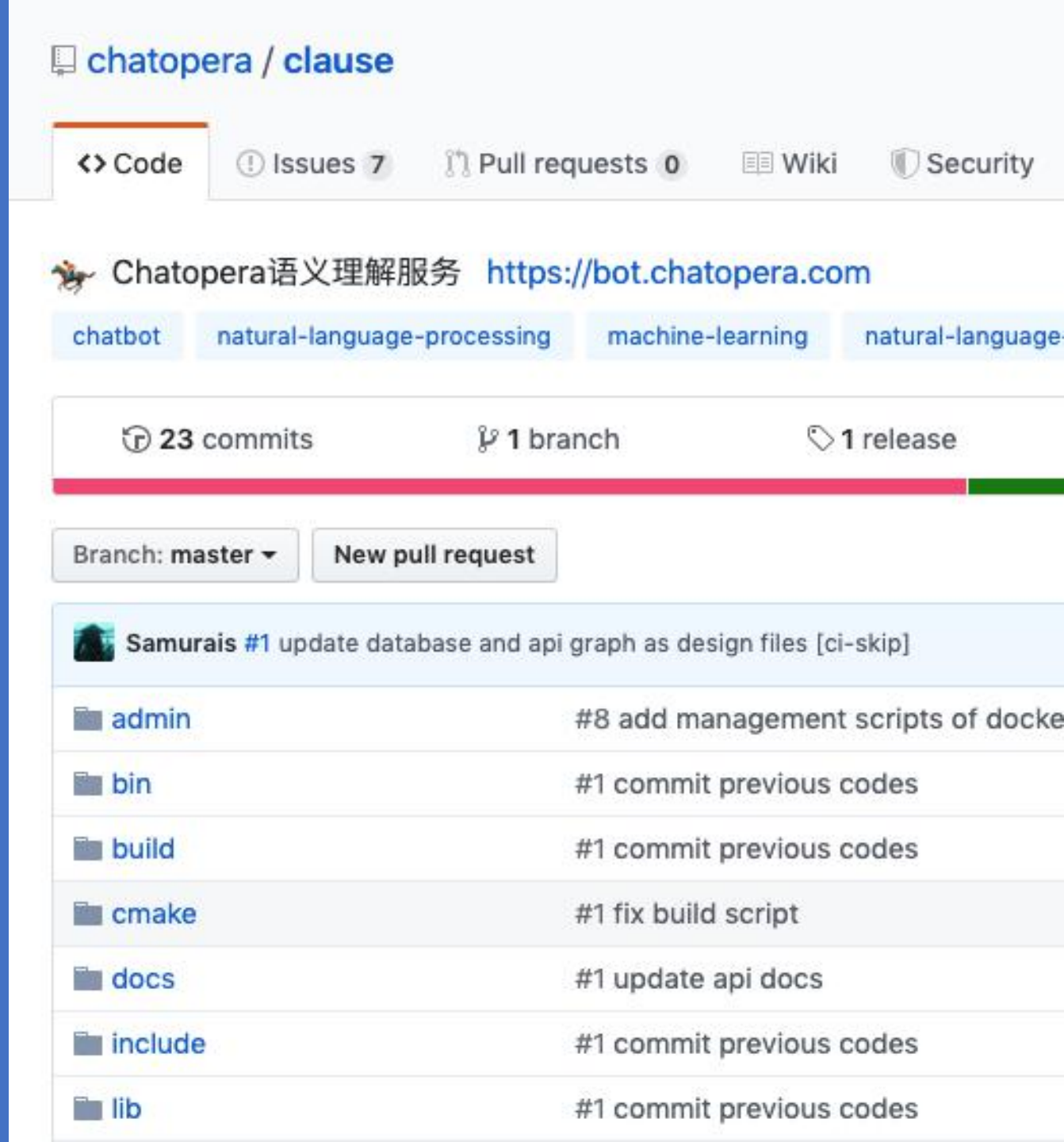
如何落地企业智能问答机器人呢？

工程路线

语义理解系统 Clause

Clause 是帮助中小型企业快速而低成本的获得好用的语义理解服务的系统。

基于意图识别的对话



二、Clause项目的设计与实现

设计

- 主要概念
- 数据库表设计
- 微服务模块设计
- 数据结构及接口

实现

- 模型训练
- 对话检索
- 主要技术栈

主要概念

词典：机器人的词汇表

- 自定义词典
- 系统词典

意图：一个任务的最小单元

- 说法
- 槽位

主要概念

调试：从设计到实现机器人的技能

- 训练机器人
- 测试对话

版本：训练好的机器人模型

- 调试版本
- 生产版本

主要概念

会话：和用户的单个意图关联的对话

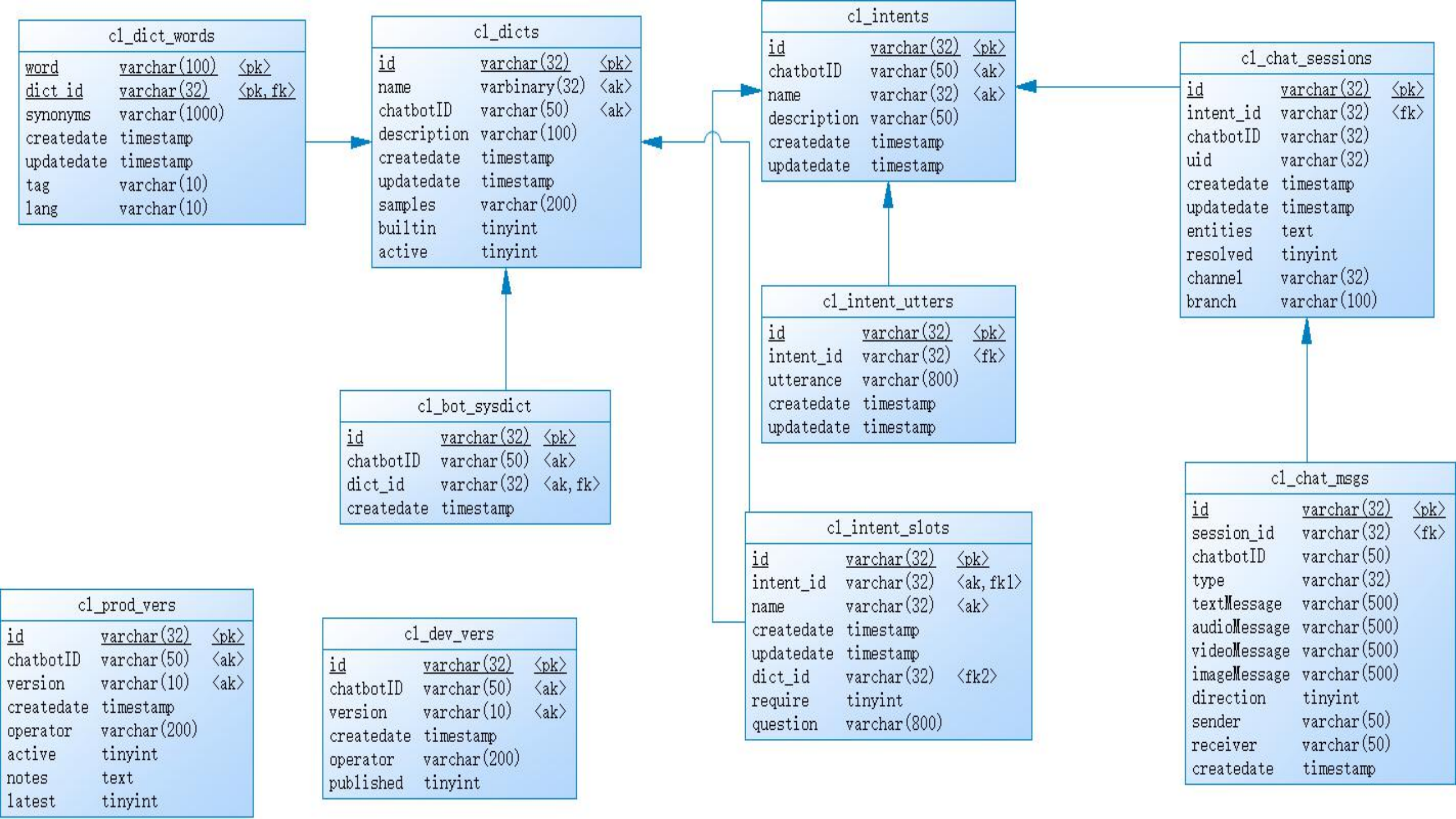
请求对话，需要先创建会话，会话会绑定0-1个任务：刚开始不知道用户意图，当确定用户意图后，该session就只和这个意图相关。

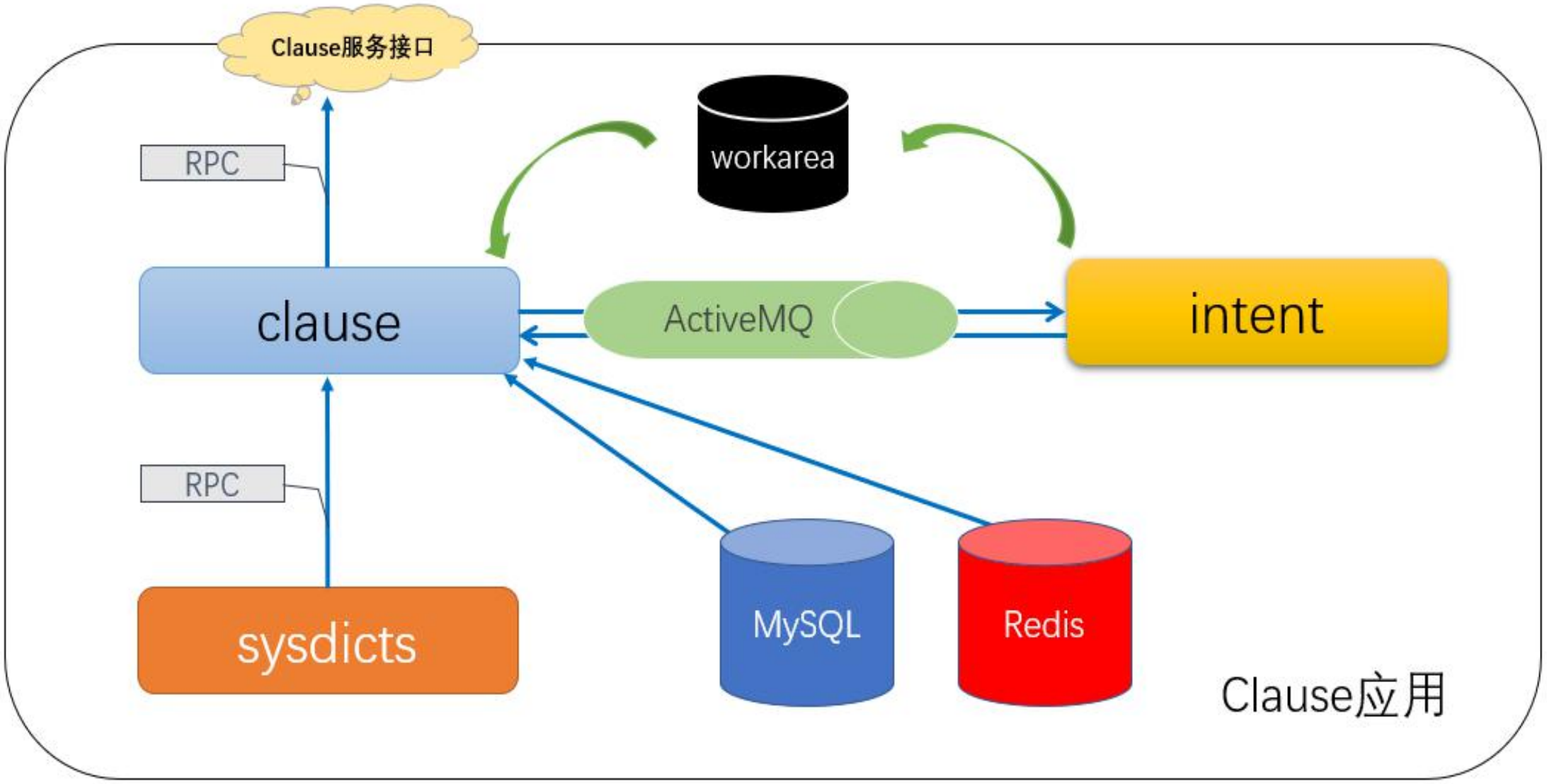
聊天：发送访客的请求，获得机器人的回复

通过访客的ID，session id, 文本消息和机器人进行聊天

表	描述	表	描述
cl_dicts	词典，包括自定义词典和系统词典	cl_chat_sessions	会话
cl_dictwords	自定义词典的词条	cl_chat_msgs	对话消息
cl_intents	意图	cl_bot_sysdict	机器人关联系统词典表
cl_intent_slots	意图槽位	cl_prod_vers	生产版本
cl_intents_utters	意图说法	cl_dev_vers	调试版本

数据库表设计


cl_dict_words
word varchar(100) <pk>
dict_id varchar(32) <pk,fk>
synonyms varchar(1000)
createdate timestamp
updatedate timestamp
tag varchar(10)
lang varchar(10)
cl_dicts
id varchar(32) <pk>
name varbinary(32) <ak>
chatbotID varchar(50) <ak>
description varchar(100)
createdate timestamp
updatedate timestamp
samples varchar(200)
builtin tinyint
active tinyint
cl_intents
id varchar(32) <pk>
chatbotID varchar(50) <ak>
name varchar(32) <ak>
description varchar(50)
createdate timestamp
updatedate timestamp
cl_chat_sessions
id varchar(32) <pk>
intent_id varchar(32) <fk>
chatbotID varchar(32)
uid varchar(32)
createdate timestamp
updatedate timestamp
entities text
resolved tinyint
channel varchar(32)
branch varchar(100)
cl_intent_utters
id varchar(32) <pk>
intent_id varchar(32) <fk>
utterance varchar(800)
createdate timestamp
updatedate timestamp
cl_bot_sysdict
id varchar(32) <pk>
chatbotID varchar(50) <ak>
dict_id varchar(32) <ak,fk>
createdate timestamp
cl_chat_msgs
id varchar(32) <pk>
session_id varchar(32) <fk>
chatbotID varchar(50)
type varchar(32)
textMessage varchar(500)
audioMessage varchar(500)
videoMessage varchar(500)
imageMessage varchar(500)
direction tinyint
sender varchar(50)
receiver varchar(50)
createdate timestamp
cl_intent_slots
id varchar(32) <pk>
intent_id varchar(32) <ak,fk1>
name varchar(32) <ak>
createdate timestamp
updatedate timestamp
dict_id varchar(32) <fk2>
require tinyint
question varchar(800)
cl_prod_vers
id varchar(32) <pk>
chatbotID varchar(50) <ak>
version varchar(10) <ak>
createdate timestamp
operator varchar(200)
active tinyint
notes text
latest tinyint
cl_dev_vers
id varchar(32) <pk>
chatbotID varchar(50) <ak>
version varchar(10) <ak>
createdate timestamp
operator varchar(200)
published tinyint


Clause服务接口
RPC
workarea
clause
ActiveMQ
intent
RPC
sysdicts
MySQL
Redis
Clause应用

微服务模块设计

数据结构与接口：词典相关

接口	描述	接口	描述
postCustomDict	创建自定义词典	unrefSysDict	取消引用系统词典
putCustomDict	更新自定义词典	myDicts	获得指定机器人的所有自定义词典和引用词典
getCustomDict	获得自定义词典详情	mySysDicts	指定机器人引用的系统词典
delCustomDict	删除自定义词典	putDictWord	创建或更新自定义词条
getSysDicts	获得系统词典列表	delDictWord	删除自定义词条
refSysDict	引用系统词典	hasDictWord	检查一个词条是否在自定义词典中
postSysDict	创建系统词典	getDictWords	获得自定义词典的词条

数据结构与接口：意图相关

接口	描述	接口	描述
postIntent	创建新的意图	delSlot	删除槽位
putIntent	更新意图	postUtter	创建意图说法
getIntents	获得意图列表	putUtter	更新意图说法
delIntent	删除意图	delUtter	删除意图说法
postSlot	创建意图槽位	getUtter	获得意图说法详情
putSlot	更新意图槽位		
getSlots	获得意图槽位列表		

数据结构与接口：训练、版本和对话

子模块	接口	描述	子模块	接口	描述
训练机器人			对话		
	train	训练机器人		putSession	创建会话
	status	获得机器人训练状态		getSession	获得会话详情
版本管理				chat	与指定的机器人进行对话
	devver	获得最新调试版本信息			
	prover	获得最新生产版本信息			
	online	将指定调试版本升级为生产版本			

数据结构

Clause暴露的集成接口的
输入和输出采用统一的数据类型:

`chatopera::bot::clause::Data`

数据结构及接口文档
https://dwz.cn/jHI09wV2

Data字段	描述	Data字段	描述
rc	返回值的代码，成功返回则rc=0,否则为异常返回	error	异常返回值的原因，当成为返回值，error为空
msg	成功返回的消息	chatbotID	聊天机器人的唯一标识
customdicts	自定义词典列表	sysdicts	系统词典列表
botsysdicts	机器人引用的系统词典列表	dictwords	自定义词典的词条列表
messages	聊天消息列表	intents	意图列表
slots	意图槽位列表	utters	意图说法列表
…			

模型训练的实现

自定义词典存储

- Hat-tire 前缀树空间
- Leveldb 词汇存储

系统词典

- 基于独立模块，RPC连接
- 使用Baidu LAC开源软件，实现LOC, ORG, PER, TIME
- 添加更多，按照插件思路设计

分类意图

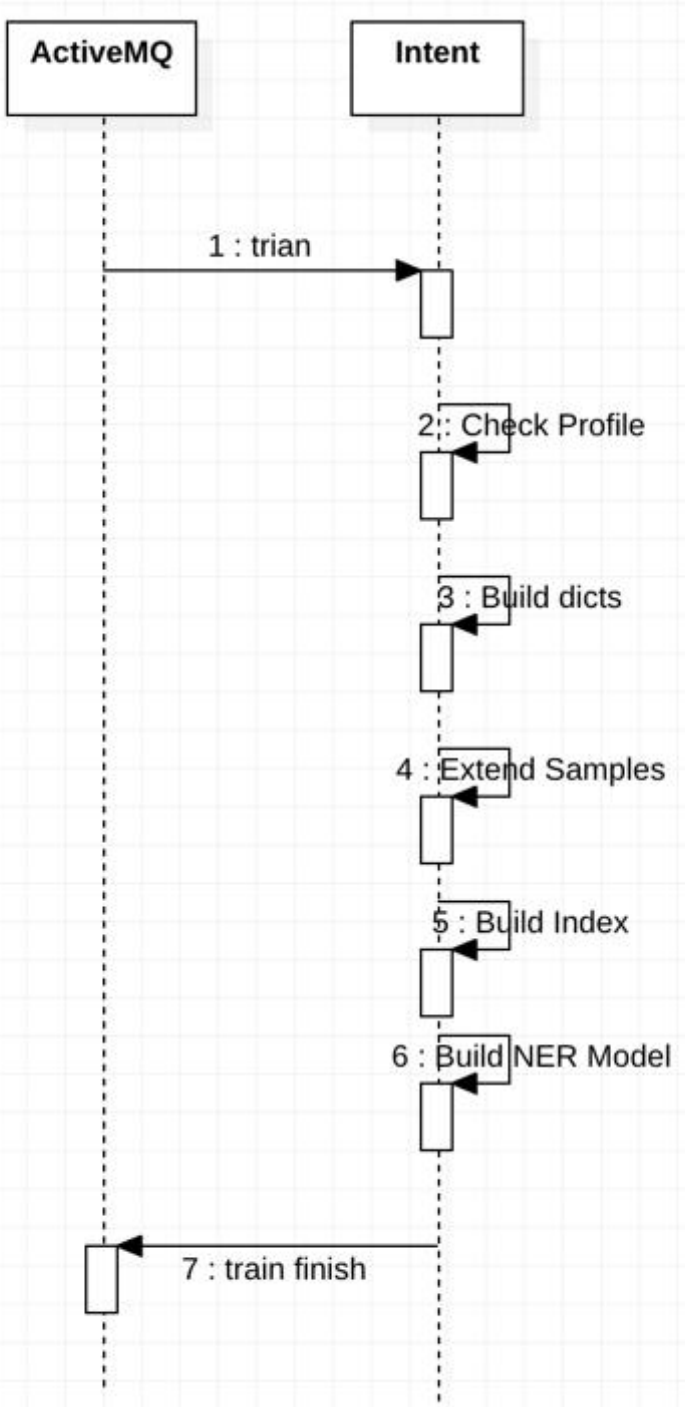
- 使用槽位的绑定的词典，生成不同说法的笛卡尔集合
- 训练生成索引，使用Xapian完成

命名实体标识

- 使用CRF算法进行序列标注和预测标签
- 训练NER模型，使用Crfsuite完成

模型训练的实现

1. 分发训练任务
2. 检查工作内容
3. 构建词典
4. 生成拓展的说法
5. 基于说法构建索引
6. 构建NER模型
7. 返回训练执行结果



模型训练的实现

augmented.json

crfsuite.ner.model

crfsuite.train.txt

dictwords.trie.bin

leveldb

xapian

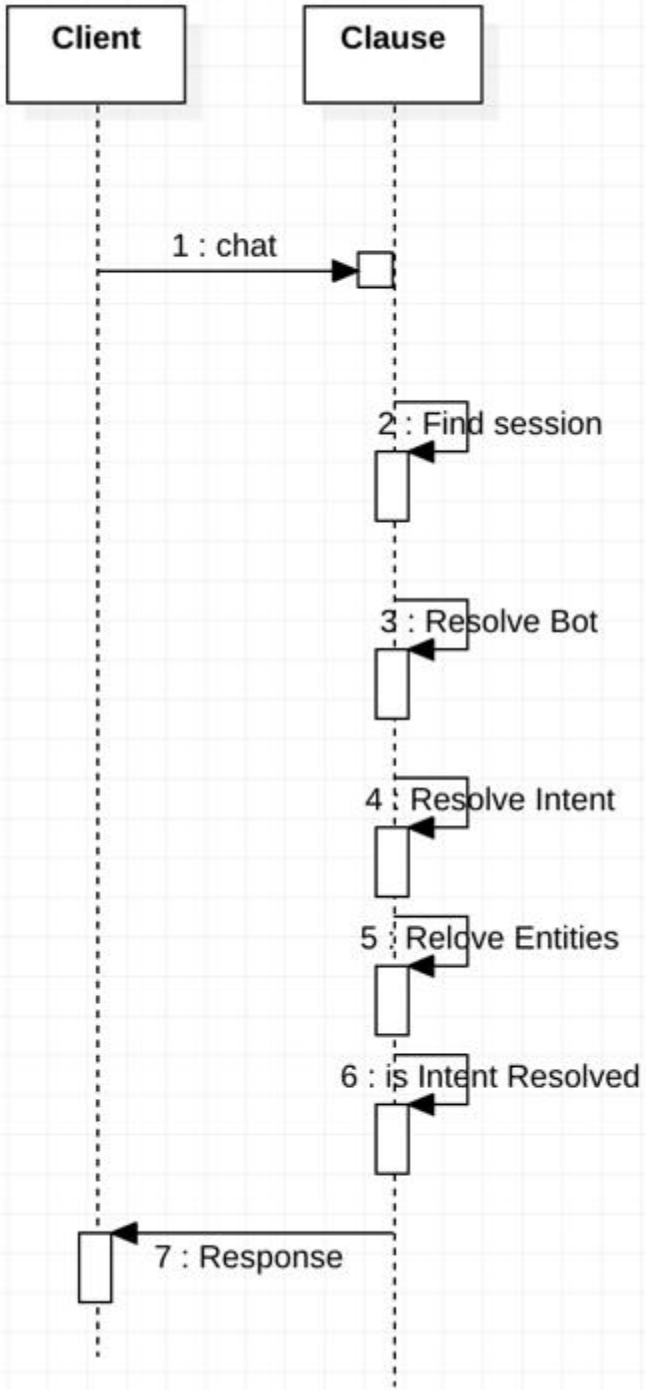
profile.json

profile.pbs

```
var/local/workarea/chatbot_id/develop
├── augmented.json
├── crfsuite.ner.model
├── crfsuite.train.txt
├── dictwords.trie.bin
├── jieba
│   ├── hmm_model.utf8
│   ├── idf.utf8
│   ├── jieba.dict.utf8
│   ├── pos_dict
│   ├── README.md
│   ├── stop_words.utf8
│   └── user.dict.utf8
├── leveldb
│   └── MANIFEST-000004OG
├── profile.json
├── profile.pbs
└── xapian
    ├── docdata.glass
    ├── flintlock
    ├── iamglass
    ├── position.glass
    ├── postlist.glass
    └── termlist.glass
```

对话检索的实现

1. 确认会话的有效
2. 确认目标BOT存在
3. 检查是否确定了意图
4. 确认新的信息提取到了槽位信息



主要技术栈

C++	Crfsuite	LevelDB	ActiveMQ
Apache Thrift	LAC/PaddlePaddle	Hat-trie	CMake
Google Protobuf	Xapian	Jieba	MySQL/Redis

Clause© opensource

Chatopera Language Understanding Service

三、开源的Clause

开源精神

- 自由
- 友爱
- 团结
- 积极
- 互助
- 进取

OpenSource

打造最好的中文语义理解开源软件

- 专注于中文处理
- 提供高性能，高可靠性的软件
- Apache2.0 授权证书，商业友好

投入社区建设

- QQ群答疑
- 社区活动分享
- 提供商业支持：云服务、技术培训和定制化开发

项目地址

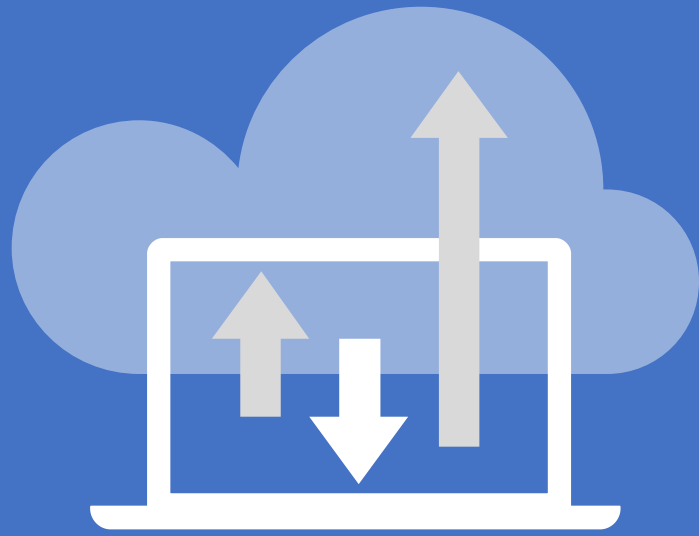
https://github.com/chatopera/clau

QQ群

群名称:Chatopera云服务
群　号:809987971

四、基于Clause快速构建聊天机器人

演示



代码片段

示例程序地址：https://github.com/chatopera/clause-py-demo

```
# 安装包
pip install clause

# 引入类
from clause import Client, Data
from clause import CustomDict, SysDict, DictWord
from clause import Intent, IntentSlot, IntentUtter
from clause import Entity, ChatMessage, ChatSession

# 实例化bot对象
bot = Client(CL_HOST, CL_PORT)
```

代码片段

```
# 创建自定义词典
data = Data()
data.customdict = CustomDict(name=customDictName,
                             chatbotID=chatbot_id)
resp = bot.postCustomDict(data)

# 更新自定义词典
data = Data()
data.chatbotID = chatbot_id
data.customdict = CustomDict(name=customDictName,
                             chatbotID=chatbot_id)
data.dictword = DictWord(word="西红柿",
                             synonyms="狼桃;柿子;番茄")
resp = bot.putDictWord(data)
```

代码片段

```
# 引用系统词典
data = Data()
data.chatbotID = chatbot_id
data.sysdict = SysDict(name="@TIME")
resp = bot.refSysDict(data)
```

代码片段

```
# 创建意图
data = Data()
data.intent = Intent(chatbotID=chatbot_id,
                     name=intent_name)
resp = bot.postIntent(data)

# 创建意图槽位
data = Data()
data.intent = Intent(chatbotID=chatbot_id, name=intent_name)
data.slot = IntentSlot(name="vegetable", requires=True,
                       question="您需要什么配菜")
data.customdict = CustomDict(chatbotID=chatbot_id,
                             name=customDictName)
resp = bot.postSlot(data)
```

代码片段

```
  # 创建意图说法
data = Data()
data.intent = Intent(chatbotID=chatbot_id,
                     name=intent_name)
data.utter = IntentUtter(utterance="帮我来一份{vegetable}, 送到
                                   {location}")
resp = bot.postUtter(data)
```

代码片段

```
  # 训练机器人
data = Data()
data.chatbotID = chatbot_id
resp = bot.train(data)

  ## 训练是一个长时间任务，进行异步反馈
while True:
    sleep(3)
    data = Data()
    data.chatbotID = chatbot_id
    resp = bot.status(data)
    if resp.rc == 0:
        break
```

代码片段

```
  # 对话
  ## 创建session
data = Data()
data.session = ChatSession(chatbotID=chatbot_id,
      uid="py", # 用户唯一的标识
      channel="testclient", # 自定义，代表该用户渠道由字母组成
      branch="dev" ) # 分支，有连个选项：dev，调试版本；pro，生产版本
sessionId = bot.putSession(data).session.id

  ## 聊天
data = Data()
data.session = ChatSession(id=sessionId)
data.message = ChatMessage(textMessage= "我想点外卖")
resp = bot.chat(data)
```

系统集成

获得SDK使用说明

发布到包管理平台，示例程序的语言	
语言	**代码地址**
Node.js	https://github.com/chatopera/node-clause
Java	https://github.com/chatopera/java-clause
Python	https://github.com/chatopera/py-clause

其他语言的客户端，在项目中的位置：
https://github.com/chatopera/clause/tree/master/var/assets/clients

目前，已经加入了的客户端语言包括：Java, Cpp, Python, Go, Php, Java, CSharp, Node.js.

五、总结

- 在线课程
- 《智能问答与深度学习 》
- Clause开源项目

资源

- 开源地址: http://github.com/chatopera
- 文档中心: https://docs.chatopera.com/
- Chatopera云服务: https://bot.chatopera.com/