

**计算机科学技术学院**

**本科生毕业论文**

**题 目：对话式智能酒店预订系统——人机交互的实现**

**专 业： 计算机科学与技术师范汉授**

**学 生： 赵瑞霞**

**学 号： 20161102908**

**指导教师： 李艳玲**

**日 期： 2020年3月10日**

**计算机科学技术学院制**

目录

[1 绪论 1](#_Toc37343446)

[1.1 研究背景和意义 1](#_Toc37343447)

[1.2 研究内容 2](#_Toc37343448)

[2 整体功能分析 2](#_Toc37343449)

[2.1 数据分析及预处理 2](#_Toc37343450)

[2.2 对话分析 3](#_Toc37343451)

[3 语义槽填充 4](#_Toc37343452)

[3.1 基于规则的语义槽填充 4](#_Toc37343453)

[3.2 基于CRF模型的槽填充 5](#_Toc37343454)

[3.2.1 对话语料设计 6](#_Toc37343455)

[3.2.2 CRF模型描述 6](#_Toc37343456)

[3.2.3 训练结果及分析 7](#_Toc37343457)

[4 对话功能设计 7](#_Toc37343458)

[4.1 对话逻辑设计 7](#_Toc37343459)

[4.2 查询功能设计 8](#_Toc37343460)

[4.3 对话生成设计 9](#_Toc37343461)

[5 功能的实现 10](#_Toc37343462)

[6 总结 11](#_Toc37343463)

[7 致谢 12](#_Toc37343464)

[参考文献 12](#_Toc37343465)

对话式智能酒店预订系统

——人机交互的实现

计算机科学技术学院 2016级计算机科学与技术 赵瑞霞 20161102908

指导老师 李艳玲 副教授

摘要 本文主要通过多轮人机对话实现用户对酒店的预订，该任务的核心部分就是语义槽的填充。本文主要采用了两种方法来完成语义槽填充，一种是基于规则的槽填充，一种是基于条件随机场（Conditional Random Field，CRF）模型的槽填充。在人机交互过程中，机器根据用户的回答获取相关语义槽的信息并填充，直到所有需要的语义槽参数获取完毕，结束对话返回查询结果。

关键词 语义槽填充；多轮交互；任务型对话

# 1 绪论

## 1.1 研究背景和意义

本次毕业设计的内容选自“第七届全国社会媒体处理大会——SMP2018中文人机对话技术评测”任务二的一部分，目的是在特定领域中通过人机多轮对话来完成用户的意图任务，本课题主要实现对酒店的预订。

目前任务型对话系统是学术界和工业界的研究热点，近年来受到了国内外学者和工业界的广泛关注。学术上，人机对话是人机交互最自然的方式之一，其发展影响及推动着语音识别与合成、自然语言理解、对话管理以及自然语言生成等研究的进展；产业上，众多产业界巨头相继推出了人机对话技术相关产品，如微软小冰、苹果Siri等，并将人机对话技术作为其公司的重点研发方向。

在科技飞速发展的今天，人们对方便快捷有越来越多的要求, 会使智能预订系统有着很大的市场需求与更大的发展空间, 为了应对用户的需求, 未来智能预订系统会向更便捷的方向发展，更具全能型[1]。目前传统互联网订购系统需要用户手动筛选信息，操作复杂，而且需要用户自己选择具体的条件选项，没有良好的交互过程。人们需要一种操作简单方便的方式来预订，即可用人机系统来处理相关信息实现预订操作。本文采用人机对话系统来降低操作的复杂度，提供简洁的界面，并保证良好的交互过程和结果反馈，通过更加自然且智能的对话系统来使用户得以简便的完成这些事情，为人们提供了更多的便利。

## 1.2 研究内容

本次毕业设计主要实现的是任务型对话系统，通过机器与用户的多轮对话，来帮助用户完成对酒店的预订。整个对话过程包括自然语言理解模块、对话管理模块和自然语言生成模块三个部分[2]。

自然语言理解是指将用户输入的句子进行分析，获取对话任务的关键信息。对话理解模块需要经过领域识别、意图识别和语义槽填充三个子模块，最主要的部分就是语义槽的填充，例如对于预订酒店的任务，城市名、地点名、价格等就是与任务相关的槽信息。本文采用两种方法来完成语义槽的填充，一种是基于规则的槽填充，另一种是基于CRF模型的槽填充。

对话管理模块包括对话状态追踪和对话策略。状态追踪包括持续获取对话的各种信息，对话管理模块根据各个语义槽的状态生成对话策略，根据当前状态生成返回相应的回复，直到所有需要的语义槽填充完成后，结束对话任务。

自然语言生成模块是指根据对话管理模块的对话策略用自然语言的方式展示给用户，给用户一个反馈结果。

# 2 整体功能分析

## 2.1 数据分析及预处理

比赛数据集是JSON格式的，其中包括一些价格为零、地址为空等不规整的数据，首先需要去除这些杂乱的数据，并将价格为零的数据根据平时酒店价格的高低，在一定范围内产生一个随机数，使其变为有效数据。

本设计将处理过的数据按照城市名切分数据，并统一编码格式。每一个城市生成一个文本文件，每一个文本文件包含的内容有城市名、酒店名、价格和地址，例如“北京, 易佰连锁旅店（石景山游乐园店）, 125, 石景山区八角东街25-13号”，这是其中的一条数据。这样做可以方便数据的读取，直接根据城市名读取相应的文本文件即可。

## 2.2 对话分析

根据日常生活中人们预订酒店的信息得知，预订一个酒店至少需要有城市名称、地点名、价格等信息，有些用户还会要求订一个特定的连锁酒店。

针对上述分析，用户可能输入的句子大体有如下几种：

1. 请帮我找一家便宜舒适的酒店
2. 我要在北京预订一个酒店
3. 我想预订一个在上海徐家汇附近（/周围/周边/旁边）的酒店
4. 订一个在北京离保福寺桥近的锦江之星酒店
5. 查找一个邻近（/靠近）上海虹桥机场的酒店
6. 帮我看看在北京积水潭附近有什么酒店
7. 我要订一个汉庭（连锁）酒店
8. 给我预订一个与北京展览路相邻的如家连锁（/快捷）酒店

当用户输入以上类似的句子时，机器获取关键的信息，进行相关问题的回复，多轮交互，完成语义槽的填充，直到所需要的语义槽填充完毕结束对话，返回查询结果，实现酒店的预订。整体框架如图1所示：

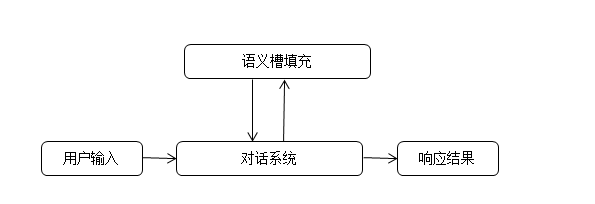


图1 整体框架图

# 3 语义槽填充

通过分析酒店预订任务，可以明确预订系统需要设计的槽包括城市名、地点、价格、酒店名等，槽填充的目标就是抽取这几个槽信息。设计的槽信息如表1所示：

表1 语义槽定义表

|  |  |  |
| --- | --- | --- |
| 语义槽名称 | 说明 | 数据类型 |
| city | 城市名 | String |
| price | 价格 | List |
| abovePrice | 价格（高于某个价格） | List |
| belowPrice | 价格（低于某个价格） | List |
| address | 地点名 | String |
| hotel | 酒店名 | String |
| require | 要求标志 | Bool |

上表中的price是用来填充比如输入的价格是“300元左右（/上下）”或者“大约（/大概）200元”或者“100-300元”这样的内容，需要指出的是必须填充的语义槽包括城市名、地点名、价格和要求标志，酒店名的填充是可选的。

## 3.1 基于规则的语义槽填充

基于规则的语义槽填充本文采用的主要是字符串匹配方法。考虑到一个城市有多个地名，并且每个城市获取语义槽信息的方法相似，所以本文只针对北京和上海这两个大城市展开研究。

首先需要弄四个列表：城市列表、北京地名列表、上海地名列表以及酒店列表，城市列表里存放的是北京和上海这两个城市，北京、上海地名列表里分别存放列举的是一些常用的地名，而酒店列表里是全国一些知名的连锁酒店名称。然后用户输入一句话进行字符串匹配，看这句话中是否有城市列表里的城市，如果问句中有相关的城市就分割出来填充到city这个语义槽，然后根据城市名再将这句话和对应的地名列表里的每个地名进行匹配，如果匹配到就将这个地名填充到address这个槽中，对于hotel语义槽也是用同样的字符串匹配的方法进行填充。

对于价格语义槽的填充采用的是根据jieba先将句子进行分词并词性标注，然后再根据某个词前后的内容进行关键词提取，填充到对应的语义槽中。本文将价格细分为三个语义槽：price、belowPrice、abovePrice。

1. price填充：
2. 根据“-”或者“到”前后的各一个词进行关键词提取，填充到语义槽；
3. 根据“左右”或者“上下”的前面的第二个词进行提取并填充；
4. 根据“大约”或者“大概”的后一个词进行提取并填充。
5. belowPrice填充：
6. 根据“不超过（/不高于/别高于/不要高于）”或者“低于”的后一个词进行提取并填充到对应的语义槽中；
7. 根据“以下”的前面的第二个词进行提取并填充。
8. abovePrice填充：
9. 根据“以上”的前面的第二个词进行提取并填充；
10. 根据“高于”的后一个词进行提取并填充。

而要求标志require的语义槽填充的类型是布尔值，如果用户有其他要求就在这个语义槽中填True，没有则在语义槽中填写False。

## 3.2 基于CRF模型的槽填充

本文将语义槽填充任务转化为序列标注任务，而序列标注就是在给定内容中对每一个字或词打上标签，再将标注结果提取进行相应语义槽的填充。本文采用的是条件随机场CRF模型实现对城市名、地点名及酒店名三个语义槽的填充。

## 3.2.1 对话语料设计

对于输入的一句话，一行一个字，句子和句子之间用空格区分，采用BIO的形式进行标注，B表示词开头begin，I表示词中间inter，O表示其他非实体，BI后面跟实体的类别。标注的示例如下图2所示：

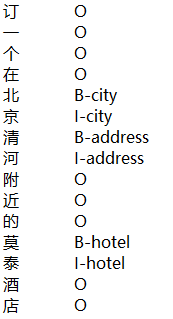


图2 CRF训练数据标注示例

## 3.2.2 CRF模型描述

本文采用CRF++工具包实现条件随机场模型[3]。CRF++是著名的条件随机场的开源工具，也是目前综合性能最佳的CRF工具。

使用CRF++工具包，首先需要定义特征模板放到template文件中，每一条模板将在每一个token处生产若干个特征函数。CRF++提供两种模板Unigram和Bigram，Unigram的特征前使用字母U，而Bigram的特征前使用字母B，后面的数字用于区分特征，模板的定义举例说明：以上图2所示的语料为例，假设当前行为“北”字这一行，那么其中一条的特征模板“U00:%x[-2,0]”代表的特征就是“个”。然后将训练的语料标注好之后放到train.txt中，经过命令“crf\_learn template train.txt model”训练数据生成模型model文件，而“crf\_test -m model test.txt >result.txt”命令是通过训练得到的模型进行测试，输出的结果是带有标注的在result文件中。

## 3.2.3 训练结果及分析

训练过程中会输出一些如下图3所示的信息：

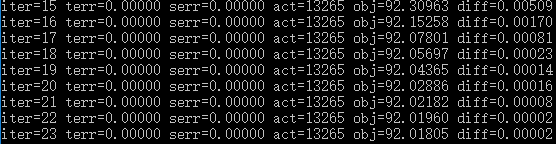


图3 训练过程中的部分内容

其意义如下：

iter：迭代次数。当迭代次数达到maxiter时，迭代终止；

terr：标记错误率；

serr：句子错误率；

obj：当前对象的值。当这个值收敛到一个确定值的时候，训练完成；

diff：与上一个对象值之间的相对差。当此值低于eta时，训练完成。

说明：通过测试发现训练结果准确率挺高，适用于本文使用。

# 4 对话功能设计

## 4.1 对话逻辑设计

整体对话逻辑设计流程图如下图4所示：

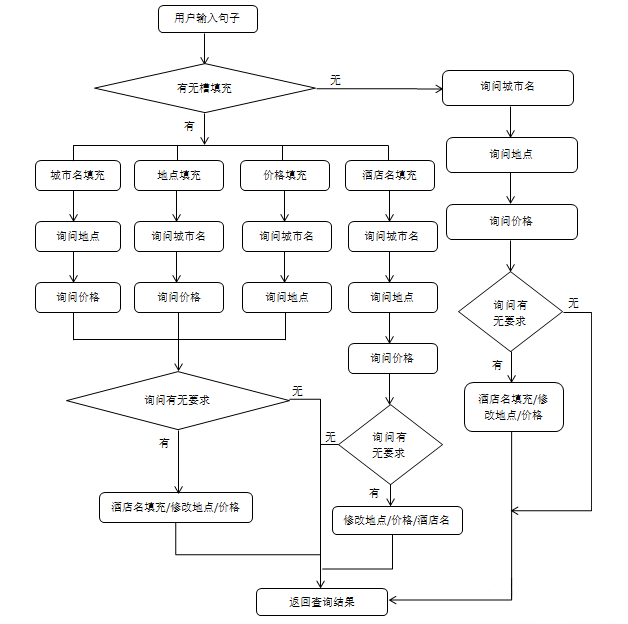


图4 对话逻辑流程图

需要指出的是，上图4所示的流程图是指用户输入的句子是简单的，一次只说明一个关键信息，例如“我要订一个在北京的酒店”这样的简单句子，实际上本系统也实现了包含多个关键信息的复杂句子，该流程图在此处省略。

## 4.2 查询功能设计

根据城市名的语义槽填充内容读取对应数据文件，然后进行模糊对比查询，将查询结果存到一个列表中，查询结束后返回查询结果。

价格查询设计：由于有三个关于价格的语义槽填充，所以需要根据相应的语义槽填充进行查询。

* 1. price查询设计：若price的长度为1，即用户输入的价格如“200元左右”，则自己规定一个范围，本文中规定的是价格上下浮动30元，在这个范围内的价格均属于满足用户的需求；若price的长度为2，即用户输入的价格在明确的两个数之间，则在数据文本文件中将满足条件的数据添加到列表中。
  2. belowPrice查询设计：将低于满足用户输入的价格放到列表中。
  3. abovePrice查询设计：将高于满足用户输入的价格添加到列表中。

地点名查询设计：在满足价格的结果列表中，查询数据中是否包含用户指出的地点名，满足就将数据放到另一个列表中，如果用户对酒店名没有具体要求，则返回查询结果即可。

酒店名查询设计：在满足价格和地点名的结果列表中，查询数据中的酒店名是否有用户所指定的，有则添加到新列表中，返回查询结果。

## 4.3 对话生成设计

根据当前语义槽的填充情况返回相应的回复。

1. 如果城市名没有填充，则询问用户“请问您要订哪个城市呢？”；
2. 其他槽有填充而地点名没有填充，则询问用户“好的，您要具体订在哪附近的呢？”；
3. 其他槽有填充而价格没有填充，则询问用户“好的，您能接受的价位是多少呢？”；
4. 城市名、地点名、价格都填充后，则询问用户“您还有什么要求吗？（如指定连锁酒店名称，修改价格、地点等）”；
5. 如果用户回答“无（/没有/没了）”，则返回“好的，查询结果如下：”；用户回答有其他要求的话，修改对应语义槽内容，然后返回语句“好的，查询结果如下：”。

# 5 功能的实现

酒店预订系统实现的人机交互的部分示例如下图5所示：

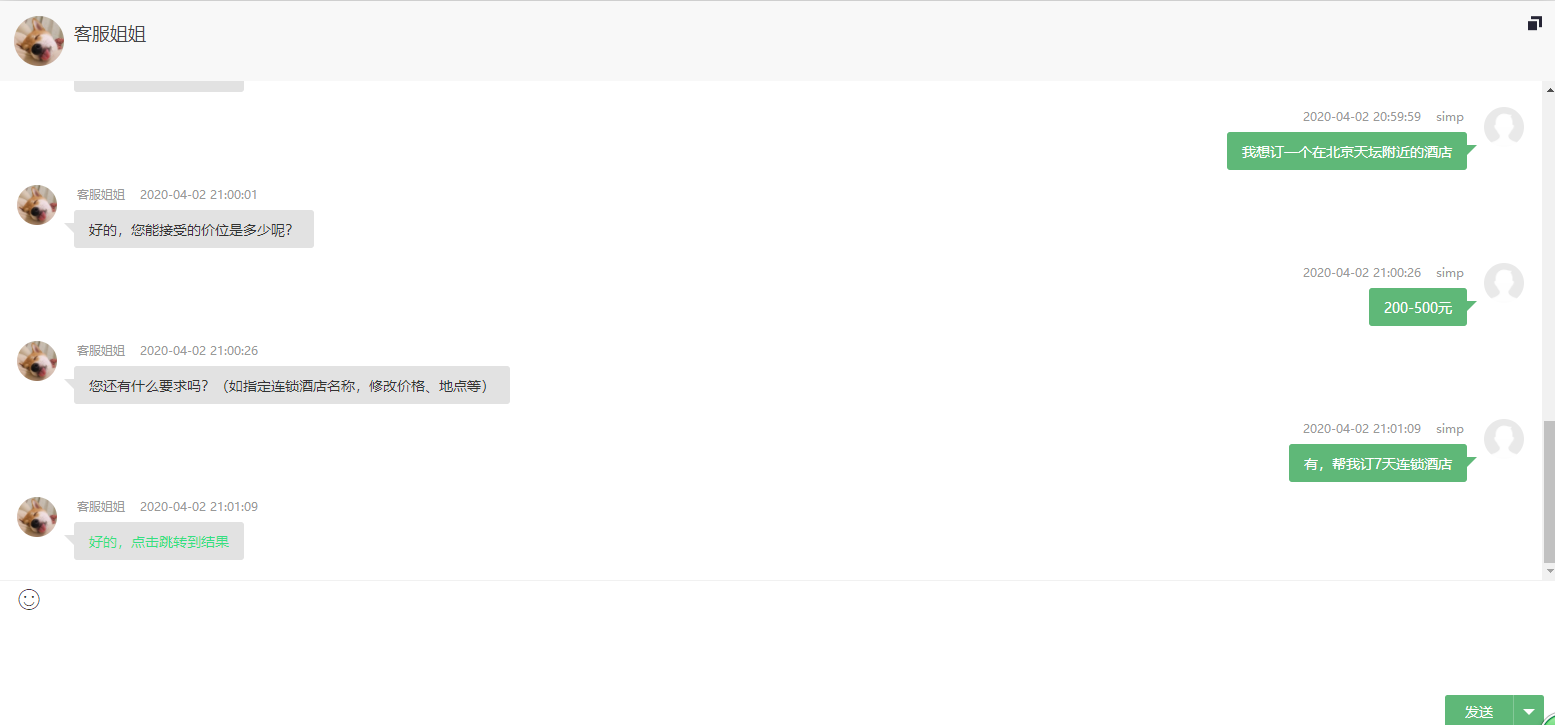


图5 人机对话过程示例

点击显示查询结果如下图6所示：

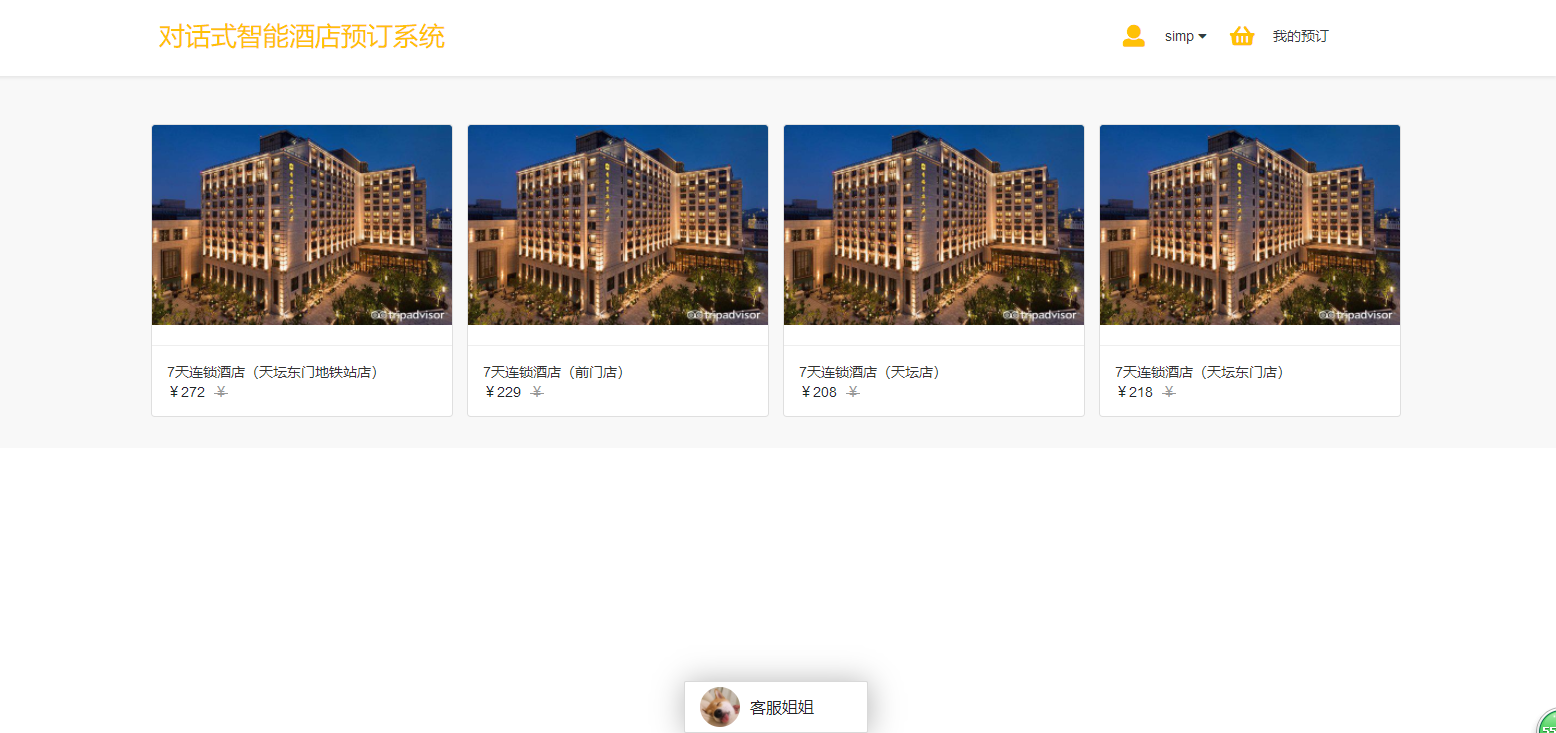


图6 显示查询结果

在显示的结果中任意选择一个心仪的酒店，然后点击添加到预订，如下图7所示：

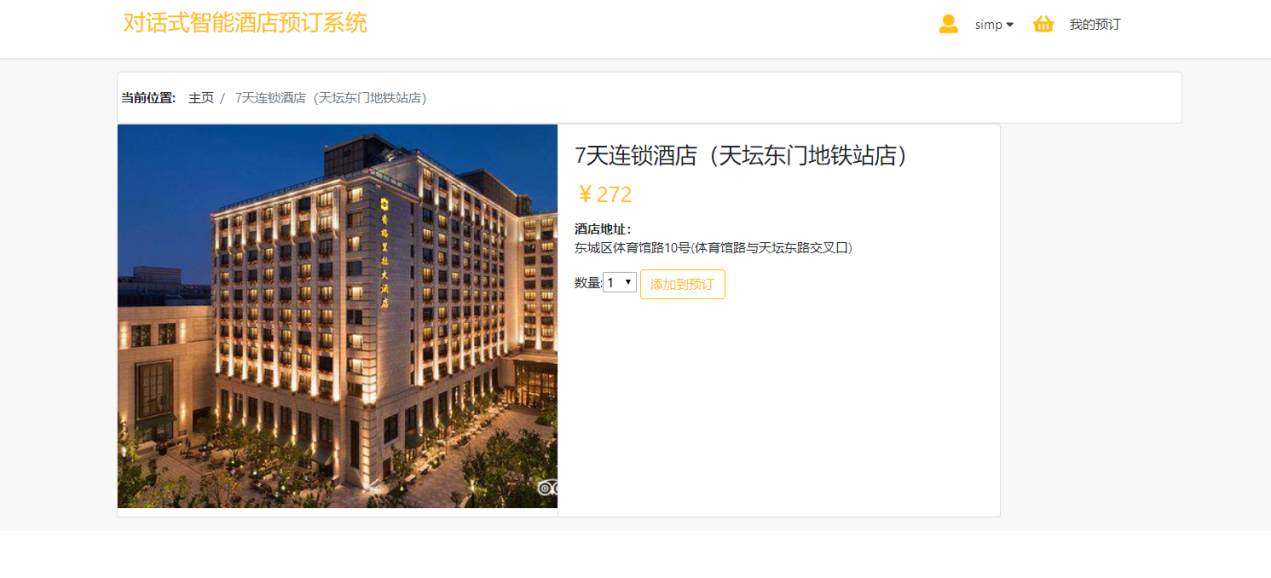


图7 添加预订

添加预订之后会出现订单信息如下图8所示：



图8 订单信息

然后点击去结算，完成预订。

# 6 总结

目前，本系统实现的是主要针对北京和上海这两个城市的酒店的预订，性能比较单一，其实还应该改善一些地方，例如自然语言生成部分也应该实现用基于深度学习的方法，这样会使系统更加智能化；交互性更好；再比如预订成功之后还可以给用户展示酒店的样式、位置、联系电话等等，使系统完善性更好，为用户带来更便利的服务。

在本次毕业设计中，我学习使用了python语言、CRF++工具包等内容，对任务型人机对话有了更深入的了解，在这个信息化的时代，智能化会越来越方便人们的生活，给人们的生活增添色彩。

# 7 致谢

经过几个月的学习，在老师、同学们的帮助下，我和同伴共同完成了对话式智能酒店预订系统。在这个过程中遇到了许多问题，通过解决这些问题锻炼了我学习的能力、增强了我的逻辑思维、开阔了我的视野，对我人生道路的指引有很大的帮助。

在这里，由衷地感谢李艳玲老师对我的耐心指导，在我遇到困难时给我指点明路，同时还要感谢学长、学姐和同伴对我的鼓励与帮助，让我受益匪浅。

# 参考文献

1. 王文昊.智能订票系统中的人机对话技术[J].科技传播，2019，11(04)：153-155.
2. 赵晗.任务型人机对话中槽填充技术的研究与应用[D].北京邮电大学，2017.
3. 林先辉.面向出行领域的任务型对话系统研究[D].哈尔滨工业大学，2018.

**Dialogue intelligent hotel reservation system**

**——The realization of human-computer interaction**

Computer Science Technical College 2016 Computer science and technology Zhao Ruixia 20161102908

Directed by Li Yanling Associate Professor

**Abstract** In this paper, the user's reservation to the hotel is realized through multiple rounds of man-machine dialogue. The core of this task is to fill the semantic slot. This paper mainly uses two methods to complete semantic slot filling, one is rule-based slot filling, the other is based on Conditional Random Field (CRF) model slot filling. In the process of human-computer interaction, the machine obtains the information of the relevant semantic slot according to the user's answer and fills it until all the required parameters of the semantic slot are obtained, and the dialogue ends to return the query results.

**Keywords** Semantic slot filling; Multi round interaction; Task-based dialogue