NLP 学习总结报告

2020 寒假人工智能项目总结

姓 名：熊勉

学 校：武汉科技大学

专 业：软件工程

2020 年 3 月 6 日

目录

1 项目背景知识

1.1 自然语言处理技术

自然语言处理（NLP）是计算机科学领域与人工智能领域中的一个重要方向。它研究能实现人与计算机之间用自然语言进行有效通信的各种理论和方法。自然语言处理是一门融语言学、计算机科学、数学于一体的科学。因此，这一领域的研究将涉及自然语言，即人们日常使用的语言，所以它与语言学的研究有着密切的联系，但又有重要的区别。自然语言处理并不是一般地研究自然语言，而在于研制能有效地实现自然语言通信的计算机系统，特别是其中的软件系统。因而它是计算机科学的一部分。自然语言处理是计算机科学，人工智能，语言学关注计算机和人类（自然）语言之间的相互作用的领域。

用自然语言与计算机进行通信，这是人们长期以来所追求的。因为它既有明显的实际意义，同时也有重要的理论意义：人们可以用自己最习惯的语言来使用计算机，而无需再花大量的时间和精力去学习不很自然和习惯的各种计算机语言；人们也可通过它进一步了解人类的语言能力和智能的机制。

实现人机间自然语言通信意味着要使计算机既能理解自然语言文本的意义，也能以自然语言文本来表达给定的意图、思想等。前者称为自然语言理解，后者称为自然语言生成。因此，自然语言处理大体包括了自然语言理解和自然语言生成两个部分。历史上对自然语言理解研究得较多，而对自然语言生成研究得较少。但这种状况已有所改变。

无论实现自然语言理解，还是自然语言生成，都远不如人们原来想象的那么简单，而是十分困难的。从现有的理论和技术现状看，通用的、高质量的自然语言处理系统，仍然是较长期的努力目标，但是针对一定应用，具有相当自然语言处理能力的实用系统已经出现，有些已商品化，甚至开始产业化。典型的例子有：多语种数据库和专家系统的自然语言接口、各种机器翻译系统、全文信息检索系统、自动文摘系统等。

自然语言处理，即实现人机间自然语言通信，或实现自然语言理解和自然语言生成是十分困难的。造成困难的根本原因是自然语言文本和对话的各个层次上广泛存在的各种各样的歧义性或多义性。

一个中文文本从形式上看是由汉字（包括标点符号等）组成的一个字符串。由字可组成词，由词可组成词组，由词组可组成句子，进而由一些句子组成段、节、章、篇。无论在上述的各种层次：字（符）、词、词组、句子、段，……还是在下一层次向上一层次转变中都存在着歧义和多义现象，即形式上一样的一段字符串，在不同的场景或不同的语境下，可以理解成不同的词串、词组串等，并有不同的意义。一般情况下，它们中的大多数都是可以根据相应的语境和场景的规定而得到解决的。也就是说，从总体上说，并不存在歧义。这也就是我们平时并不感到自然语言歧义，和能用自然语言进行正确交流的原因。但是一方面，我们也看到，为了消解歧义，是需要极其大量的知识和进行推理的。如何将这些知识较完整地加以收集和整理出来；又如何找到合适的形式，将它们存入计算机系统中去；以及如何有效地利用它们来消除歧义，都是工作量极大且十分困难的工作。这不是少数人短时期内可以完成的，还有待长期的、系统的工作。

以上说的是，一个中文文本或一个汉字（含标点符号等）串可能有多个含义。它是自然语言理解中的主要困难和障碍。反过来，一个相同或相近的意义同样可以用多个中文文本或多个汉字串来表示。

因此，自然语言的形式（字符串）与其意义之间是一种多对多的关系。其实这也正是自

然语言的魅力所在。但从计算机处理的角度看，我们必须消除歧义，而且有人认为它正是自然语言理解中的中心问题，即要把带有潜在歧义的自然语言输入转换成某种无歧义的计算机内部表示。

歧义现象的广泛存在使得消除它们需要大量的知识和推理，这就给基于语言学的方法、基于知识的方法带来了巨大的困难，因而以这些方法为主流的自然语言处理研究几十年来一方面在理论和方法方面取得了很多成就，但在能处理大规模真实文本的系统研制方面，成绩并不显著。研制的一些系统大多数是小规模的、研究性的演示系统。

目前存在的问题有两个方面：一方面，迄今为止的语法都限于分析一个孤立的句子，上下文关系和谈话环境对本句的约束和影响还缺乏系统的研究，因此分析歧义、词语省略、代词所指、同一句话在不同场合或由不同的人说出来所具有的不同含义等问题，尚无明确规律可循，需要加强语用学的研究才能逐步解决。另一方面，人理解一个句子不是单凭语法，还运用了大量的有关知识，包括生活知识和专门知识，这些知识无法全部贮存在计算机里。因此一个书面理解系统只能建立在有限的词汇、句型和特定的主题范围内；计算机的贮存量和运转速度大大提高之后，才有可能适当扩大范围.

以上存在的问题成为自然语言理解在机器翻译应用中的主要难题，这也就是当今机器翻译系统的译文质量离理想目标仍相差甚远的原因之一；而译文质量是机译系统成败的关键。中国数学家、语言学家周海中教授曾在经典论文《机器翻译五十年》中指出：要提高机译的质量，首先要解决的是语言本身问题而不是程序设计问题；单靠若干程序来做机译系统，肯定是无法提高机译质量的；另外在人类尚未明了大脑是如何进行语言的模糊识别和逻辑判断的情况下，机译要想达到“信、达、雅”的程度是不可能的。

1.2 自然语言处理技术的发展

最早的自然语言理解方面的研究工作是机器翻译。1949 年，美国人威弗首先提出了机器翻译设计方案。20 世纪 60 年代，国外对机器翻译曾有大规模的研究工作，耗费了巨额费用，但人们当时显然是低估了自然语言的复杂性，语言处理的理论和技术均不成热，所以进展不大。主要的做法是存储两种语言的单词、短语对应译法的大辞典，翻译时一一对应，技术上只是调整语言的同条顺序。但日常生活中语言的翻译远不是如此简单，很多时候还要参考某句话前后的意思。

大约 90 年代开始，自然语言处理领域发生了巨大的变化。这种变化的两个明显的特征是：

（1）对系统输入，要求研制的自然语言处理系统能处理大规模的真实文本，而不是如以前的研究性系统那样，只能处理很少的词条和典型句子。只有这样，研制的系统才有真正的实用价值。

（2）对系统的输出，鉴于真实地理解自然语言是十分困难的，对系统并不要求能对自然语言文本进行深层的理解，但要能从中抽取有用的信息。例如，对自然语言文本进行自动地提取索引词，过滤，检索，自动提取重要信息，进行自动摘要等等。

同时，由于强调了“大规模”，强调了“真实文本”，下面两方面的基础性工作也得到了重视和加强。

（1）大规模真实语料库的研制。大规模的经过不同深度加工的真实文本的语料库，是研究自然语言统计性质的基础。没有它们，统计方法只能是无源之水。

（2）大规模、信息丰富的词典的编制工作。规模为几万，十几万，甚至几十万词，含有丰富的信息（如包含词的搭配信息）的计算机可用词典对自然语言处理的重要性是很明显的。

1.3 项目要求

设计一个采用问询股票信息为应用背景的金融聊天机器人。

2 项目相关知识介绍

2.1 Telegram

Telegram 是在美国比较常用的跨平台的即时通信软件，其客户端是自由及开放源代码软件，但服务器是专有软件。用户可以相互交换加密与自毁消息，发送照片、影片等所有类型文件。官方提供手机版、桌面版和网页版等多种平台客户端；同时官方开放应用程序接口（API），因此拥有许多第三方的客户端可供选择，其中多款内置中文。正是由于 Telegram 对外开放程序接口的缘故，使得 Telegram 成为实现程序界面的一个非常重要的工具。

要将 python 项目和 Telegram 集成，需要下载 python-telegram-bot 库，它是围绕 Telegram Bot API 的 Python 包装器，该库为 Telegram Bot API 提供了一个纯 Python 接口。它适用于 2.6+ 及更高版本的 Python。它还可以与 Google App Engine 一起使用。

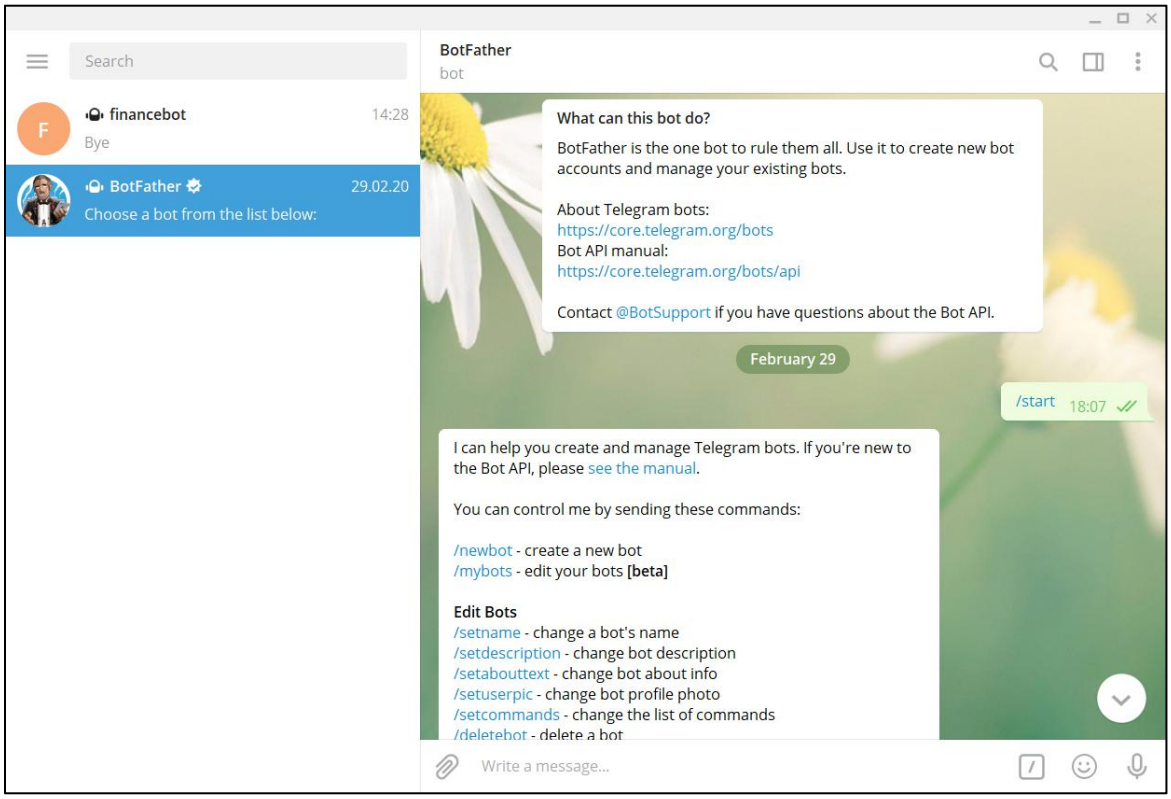
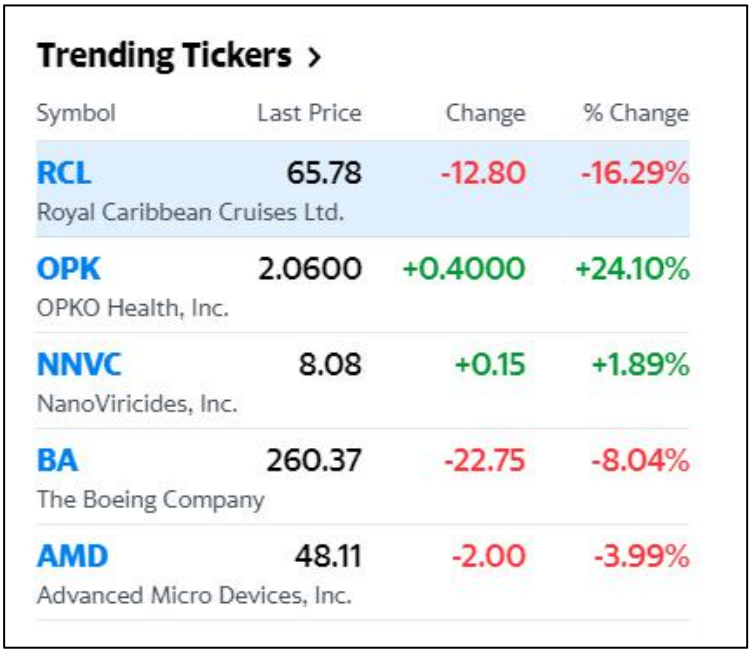


图 1 创建新的机器人

2.2 Yahoo Finance

在研究有关股市领域的项目时，Yahoo Finance 是一个非常好用的工具。它可以提供超过 5 年的每日 OHLC 价格数据，而且它是免费且可靠的。

有一个新的 python 模块 yfinance 可以包装新的 Yahoo Finance API，可以在项目中直接使用它。和其他工具例如 iexfinance 或是 googlefinance 相比，它不仅提供免费 API，而且在获取数据时也是非常方便且数据格式易于处理。这也是我之所以选择 Yahoo Finance 来实现这个项目的重要原因，利用 yfinance 库可以较快速获取五年之内甚至以上的 OHLC 数据。



Trending Tickers ›

Symbol	Last Price	Change	% Change
RCL Royal Caribbean Cruises Ltd.	65.78	-12.80	-16.29%
OPK OPKO Health, Inc.	2.0600	+0.4000	+24.10%
NNVC NanoViricides, Inc.	8.08	+0.15	+1.89%
BA The Boeing Company	260.37	-22.75	-8.04%
AMD Advanced Micro Devices, Inc.	48.11	-2.00	-3.99%

图 2 Yahoo Finance 主页的 Trending Tickers

Stocks: Most Actives ›

Symbol	Last Price	Change	% Change
BAC Bank of America Corporation	26.78	-1.43	-5.07%
GE General Electric Company	10.08	-0.87	-7.95%
AMD Advanced Micro Devices, Inc.	48.11	-2.00	-3.99%
F Ford Motor Company	6.74	-0.34	-4.80%
NIO NIO Limited	3.7200	-0.1500	-3.88%

图 3 Yahoo Finance 主页的 Most Active Stocks

3 知识点总结

3.1 Rasa NLU

Rasa NLU 是一种开源自然语言处理工具，用于聊天机器人中的意图分类，响应检索和实体提取。Rasa NLU 曾经是一个单独的库，但现在是 Rasa 框架的一部分。使用 Rasa NLU 可以通过提供配置文件以及训练数据的 json 数据格式的文件来训练一个想要实现功能的数据模型。这个模型在训练数据集规模达到一定程度时，可以较理想地解析一个句子，分析出句子的各种成分。相反地，如果训练数据集较小，则系统会发出警告，所训练得出的模型也是相当不准确的，也不能够按我们所预期的准确地解析一个句子，因此，对于 Rasa NLU 来说，训练数据集的规模是非常重要的，需要考虑到各种输入情况。此外，根据项目实际情况选择合适的 pipeline 也是很重要的，比如两个最重要的管道是 supervised_embeddings 和 pretrained_embeddings_spacy。两者最大的区别是 pretrained_embeddings_spacy 管道使用来自 Glove 或者 fastText 预训练词向量。另一方面，supervised_embeddings 管道，并不使用任何预训练的词向量，而是拟合你自己指定的数据。

运用到项目中时，pretrained_embeddings_spacy 管道的优点是，如果有一个训练样本，例如：”I want to buy apples”，并且要求 Rasa 预测”get pears”的意图，模型已经知道“苹果”和“梨”非常相似。如果没有很多训练数据，这将特别有用；supervised_embeddings 管道的优点是可以为 domain(域)自定义单词向量。例如，在一般英语中，”balance”一词与"symmetry”密切相关，但与”cash”一词非常不同。在银行业领域，“balance”和“cash”密切相关，如果希望模型能够捕捉到这一点则 supervised_embeddings 管道是一个不错的选择。

3.2 选择性回答

在设计机器人时，往往预期呈现出同一个问题的回答方式是灵活的，就类似人与人交流一样，同一个问题的回答是多变的。要想实现选择性回答，可以设置一个字典，键存放问题的意图，对应的值为相应意图的回答方式的列表。当提问的意图和字典中任何一个键匹配时，回答从值列表中任意选择一个。利用 random 库中的 choice()函数，可以从一个列表中任意选择一个值输出，因而实现了选择性回答。

3.3 正则式匹配

利用正则式可以构建许多种模式，例如：”I want an apple”,”I want a pear”,”I want a banana”这三个句子有一个共同部分，那就是”I want”这部分，这时利用正则式”I want (.*)”就可以匹配到这三句话。在这个项目中，我运用了正则式匹配去除掉了句子中一些不重要的信息，就像在人与人交流之中，并不是每一句话的每一个词是关键词，因此在处理语句的时候，完全可以忽略一些不太影响句意的单词。例如：”I wonder the volume of AAPL yesterday”这句话中”I wonder”的存在与否并不影响这句话的意图，即为”the volume of AAPL yesterday”；还有”Do you know the price of TSLA from 2020-02-01 to 2020-02-10”这句话中”Do you know”的存

在与否也不影响这句话的意图，即为查询”the price of TSLA from 2020-02-01 to 2020-02-10”，所以正则式的利用可以减少一些处理过程。要想判断一个句子和一个模式是否有匹配部分，只需要借助 re 库中的 search()就可以实现了。

3.4 意图识别

意图识别的目的为识别出语句的意图，在处理文本的时候一句话的意图能否准确识别决定了获取的信息是否准确。不同的意图识别器的训练速度，识别速度以及处理方式都各不相同。

System	Ideal Sentence count	Training Speed	Recognition Speed	Flexibility
fsticuffs	1M+	very fast	very fast	ignores unknown words
fuzzywuzzy	12-100	fast	fast	fuzzy string matching
adapt	100-1K	moderate	fast	ignores unknown words
rasaNLU	1K-100K	very slow	moderate	handles unseen words
flair	1K-100K	very slow	moderate	handles unseen words

图 4 不同意图识别器的性能比较

在意图识别中，有如下几个难点：

1. 输入不规范：不同的用户对同一诉求的表达是存在差异性的。
2. 多意图，比如查询词为：“水”，到底是矿泉水，还是女生用的化妆水。
3. 数据冷启动：当用户行为数据较少时，很难获取准确的意图。

在完成这个项目时，我使用的是 Rasa NLU 来完成意图识别的，通过编写训练数据集，利用 rasa nlu 系统训练数据，在数据量较大的情况下，是可以较好的识别出语句意图的。

3.5 命名实体和否定实体识别

命名实体识别（Named Entity Recognition，简称 NER），又称作“专名识别”，是指识别文本中具有特定意义的实体，主要包括人名、地名、机构名、专有名词等。在分析一个语句时，意图识别固然重要，命名实体识别也同样很重要。命名实体就是指一个句子中具有特定意义的实体。通常来说，命名实体的识别有三种常见的方法：正则表达式匹配法，spacy 依赖树法以及 spacy 中.ents 用法。在这个项目中，我利用的是 rasa nlu 中自带的命名实体识别，同时也用相似的方法实现否定实体识别。

3.6 多轮对话和等待状态转换

所谓单轮对话和多轮对话之间的差别，其实无非是多轮对话将要表达的信息分多次告知机器人，而要实现多轮对话，一定要把前几轮的信息同本轮所得到的信息相结合，来分析用户的意图。比如：“What’s the stock price?”,”AAPL,TSLA and SBUX”,”on April,5th”这三句话逐次

扩大信息库，从“查询股价”到“查询苹果，特斯拉和星巴克公司的股价”再到“查询苹果，特斯拉和星巴克公司在四月五日的股价”，信息量是逐渐增多的，意图是越来越精确的。因此，实现多轮对话我采用的是设置全局变量来存储消息主意图，在上面例子中主意图即为“查询股价”，只要主意图没有被修改，那么之后所得到的信息都为附加信息，如“苹果，特斯拉和星巴克公司”和“四月五日”，然而一旦收到的新消息中包含的意图和当前主意图是“平行的”，则修改当前主意图。例如，新消息为“I’d like to know the volume of AAPL on June,1th”，这时将修改主意图为“查询交易量”了。

等待状态转换是基于多轮对话中的一个概念，因为对话为多轮说明信息逐渐递增的，这时就引出了一个“状态”的概念。例如在上面的例子中，当只发出了第一条信息时，系统的状态则是处于“等待股票名称和时间段输入”的状态，在发出了第二条信息后，系统的状态修改为“等待时间段输入”的状态了，最后第三条消息发送后，系统状态修改为“查询状态”。系统可以根据系统所处的相应状态来决定处理方式。

3.7 matplotlib 绘制折线图

matplotlib 是受 MATLAB 的启发构建的。MATLAB 是数据绘图领域广泛使用的语言和工具，其语言是面向过程的。利用函数的调用，MATLAB 中可以轻松的利用一行命令来绘制直线，然后再用一系列的函数调整结果。matplotlib 有一套完全仿照 MATLAB 的函数形式的绘图接口，在 matplotlib.pyplot 模块中。

在这个项目中，每当用户查询在某一段时间内某几支股票的 OHLC 价格时，都会将相应数据绘制为折线图，从而方便用户以直观的方式看出这几支股票的变化趋势以及比较价格高低。利用 matplotlib.pyplot 模块中的 plot()函数绘图，savefig()存储所绘制的图，之后再发送给用户。

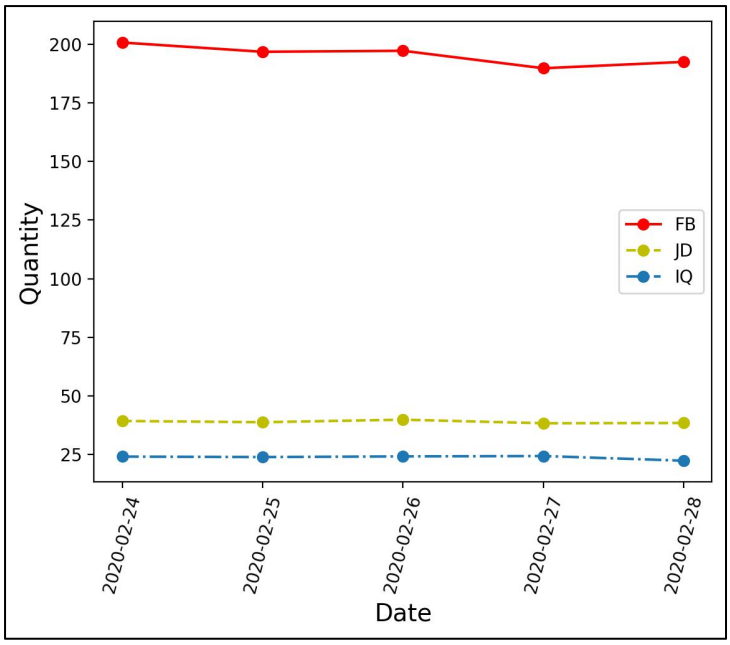


图 5 matplotlib 绘图实例

4 学习感想与收获

4.1 学习感想

这次项目虽说不是我第一次接触 nlp，但可以说第一次这么深入认真地去学习自然语言处理。在这次项目以前总是想要去学习 python 语言，但可以说每次因为各种各样的缘由没有进行下去。借着这次项目的劲头，在项目开始上课前一个多月，我开始了 python 的学习，从集合，列表，元组再到字典类型，以及如何使用它们，再到后面类的学习，我开始对 python 语法熟悉了起来，之后就是更加深入的一些知识，以及用一些实际项目来加强印象，可以说因为这次项目我终于完成踏入 python 大门 的一个目标。在第一次上课时，听到老师讲课我甚至有点小庆幸，因为自己学了 python 可以听懂老师讲的有关 python 的知识，但每次有自己不熟悉的语法，课后我都及时补充那些相关的知识并多加练习，让自己对 python 的使用更灵活，我深深感觉了这一两个月来仅仅在 python 的学习上我就收获了太多太多，从开始学习，到用 python 完成了五六百行的课设，再到完成了这次项目。

因为自己对 nlp 只是有比较有限的了解，所以刚开始的时候是抱着试一试的态度来尝试学习 nlp。要说为什么想试试 nlp，因为我了解 nlp 技术目前在世界范围内还不是那么成熟，而我的性格也是那种喜欢尝试有较大发展空间的领域，所以在了解到有这么一个项目的时候我有点心动了。在听完老师第一节讲的综述说到：第一节课是讲述一下 NLP 中的难点以及如何做一个简单的人机对话系统；第二节课是讲解“意图识别”和“命名实体识别”，老师刚提到这两个专业术语的时候我还不是特别理解，随后老师给我们举了例子说明了这两个专业词汇的意思，我开始产生了更多的兴趣；第三节课是讲解有关数据库的方法，多轮对话和否定的实现，顿时我产生非常浓厚的兴趣，因为在此前我就有接触到一些对话机器人，比如：小度，siri 之类，但是一直觉得多轮对话特别神奇，可以上下文关联一直是我想得不太明白的地方；第四节课是讲解状态机和意图不同的多轮对话。除了要对这些知识有浓厚的兴趣之外还不行，还得务实地去钻研，在学习过程中不断收获到新知识我是非常高兴的，只不过在学习中遇到一些问题经常让我头大，比如在配置环境的时候，我真的花了足足三四天去配置我的电脑，因为电脑的缘故，导致同样的配置过程也会出现各种不同的问题。在这次项目中安装 rasa 的过程我向同学们请教发现他们的电脑都没有什么问题，而我的电脑则是各种报错，这是其实就是非常考验我心理的一个时候，我花了两三天时心态其实就有点不太好了，看着别人都可以配置好，我心理有点挫败感了，但是在不断地查资料，请教老师，请教同学最后终于配置完成了。

对于参与的这次项目，我由衷得感谢我的指导老师张老师，张老师一直在耐心地为我解决问题，还有和我一起的同学们，可以经常和他们探讨问题真的很幸运，我从这次项目中学到了很多知识，在 nlp 的学习上也是一次很好的启蒙，我想在未来我很有可能会从事 nlp 的研究，因为在此前我对计算机视觉是挺感兴趣的，但这次接触了 nlp 之后发现 nlp 的研究也是一件非常有意思的事情，所以最后由衷表示对指导老师张老师和同学们的感谢。

4.2 学习收获

4.2.1 第一节课

我了解到了 NLP 的难点产生的原因，比如存在的语言的多义性，例如同一句话可以表达的意思不同；还有语言表达的灵活性，也造成处理语言成为一个难点；以及语言中的暗喻手法等，有时候连如此智能的人类都不一定能完全理解一个句子的深层含义。上述问题都是导致 nlp 发展不是如此完备的原因。

此外，我还学习了选择性回答的实现：利用 random 模块中 choice()函数来随机选择列表中的值；还学习了模式匹配：利用 re 模块中的 search()函数来判断一条 message 是否和指定模式 pattern 匹配，如果匹配则可以提取关键句，利用 group()函数可以实现提取关键句的功能。

4.2.2 第二节课

老师主要讲解了“意图识别”和“命名实体识别”，我知道了前者是识别一个句子的意图，后者是识别句子中具有特殊含义的实体。实现“意图识别”有三种常用方法：1)正则表达式方法，通过编写正则表达式来构建匹配模式；2)scikit-learn 中的最近邻分类法，即利用 cosine_similarity()函数来实现；3)支持向量机/分类器法 SVM/SVC，这个是通过训练数据和给数据贴标签得到一个模型，之后可以通过给定一个测试数据集来判断模型的准确度。

我还学到了有关句子向量，词向量和字符向量的联系与差别，句子向量能最好的反映单词之间的联系，但是数据量太大；而字符向量的数据量最小，但单词之间的关联性极差。折中来看，单词向量（词向量）的使用频率最高，无论是在数据量方面来看，还是单词之间的关联性方面来看都具有较大的优势。词向量是计算密集型的，训练词向量需要大量的数据，一旦训练完成后，高质量的词向量可供任何人使用。

词语词义的相似度可由词向量来计算，单词之间的“距离”=向量之间的角度，两个词向量的余弦相似度的绝对值越接近于 1 则两个词向量相关性越高，从而词语语义高度相关（相同或是相反），反之，绝对值越接近于 0 则两个词向量相关性越低，从而词语语义高度无关。

4.2.3 第三节课

我学习到了 python 中如何结合数据库存储数据，即利用 SQLite 与 python 结合来实现。其中在查询数据的时候一个比较简便的方式是 SQL 注入的方式，即将所有条件值存储到一个元组中，最后在查询语句中一次注入所有条件值。

还有增量过滤器的使用，即为了实现多轮对话，需要把每一轮对话收集到的信息添加存储到一个变量中，把新增的信息和之前的信息结合去比对数据，实则通过增加信息量来减小搜索范围。

此外，还有否定实体的甄别，可以搜索句子中的否定词，根据句中否定词存在的位置将句子划分为多段，从而分别对每一段子句分别进行处理。显然，非否定实体是我们需要的信息，而否定实体是我们需要排除的信息。

4.2.4 第四节课

我首次接触到了状态机的使用方法，就是把状态之间的转换用字典类型存储起来，当前状态加上一个触发事件可以转换到另一个状态同时返回一个回答语句，状态机的使用很好地解决了在对话中出现的待处理的操作。

此外，我还了解到了神经对话模型（LSTM），它是由 encoder 和 decoder 组成的。seq2seq 属于 encoder-decoder 结构的一种，其基本思想就是利用两个 RNN，一个 RNN 作为 encoder，另一个 RNN 作为 decoder。encoder 负责将输入序列压缩成指定长度的向量，这个向量就可以看成是这个序列的语义，这个过程称为编码。

最后，我还学习到了 api 的使用方法，在此前我还不能熟练地使用 api，但经过老师的讲解，我开始能较熟练地使用 api，试着获取 api token，然后去调用 api 提供的各种函数。