硕士学位论文

基于 CNN 语义匹配的自动问答系统构建方法研究

RESEARCH ON CONSTRUCTING AUTOMATIC QUESTION ANSWERING SYSTEM BASED ON CONVOLUTION NEURAL NETWORK OF SEMANTIC MATCHING

邓憧

哈尔滨工业大学

2016 年 12 月

国内图书分类号：TP391.3　　学校代码：10213
国际图书分类号：621.3　　密级：　公开

工学硕士学位论文

基于 CNN 语义匹配的自动问答系统构建方法研究

硕 士 研 究 生：邓憧
导　　师：王晓龙教授
申 请 学 位：工学硕士
学　　科：计算机科学与技术
所 在 单 位：深圳研究生院
答 辩 日 期：2016 年 12 月
授予学位单位：哈尔滨工业大学

Classified Index: TP391.3
UDC: 621.3

Dissertation for the Master Degree in Engineering

RESEARCH ON CONSTRUCTING AUTOMATIC QUESTION ANSWERING SYSTEM BASED ON CONVOLUTION NEURAL NETWORK OF SEMANTIC MATCHING

Candidate: Deng Chong
Supervisor: Prof. Wang Xiaolong
Academic Degree Applied for: Master Degree in Engineering
Speciality: Computer Science and Technology
Affiliation: Shenzhen Graduate School
Date of Defence: December, 2016
Degree-Conferring-Institution: Harbin Institute of Technology

摘　要

随着信息化技术的迅速发展，用户对网络资源的获取方式也在不断的变化。从最初的黄页查取到之后的传统搜索引擎，再到现在的智能问答机器人直接获取答案，这种变化其实是计算机在自然语言理解和信息抽取方面取得的重大进展所推动的。用户越来越倾向于用更简单的方式获取信息，这也就要求计算机拥有更强的语言理解能力。近年来，越来越多的科技公司和研究机构开始进行智能问答机器人的开发，如苹果公司的智能语音助手 Siri、微软互联网工程院发布的微软小冰，而自动问答模块正是智能问答机器人中极为重要的一个模块。因此，自动问答领域已经成为了目前人工智能研究的一个热点，而在该领域各种新型机器学习方法的应用也令问答系统的智能水平不断提高。本课题的研究目的是构建一个针对常见问题的自动问答系统。

本课题的主要研究内容包括开放式的语义匹配语料集的构建、语义匹配算法设计、自动问答系统构建。针对目前在语义匹配领域还没有一个开放的中文语料集，而目前使用最广泛的 MSRP 语料集又存在数据量较小的缺点，本课题构建了一个使用于中文问句匹配的开放式语义匹配语料集。基于构建的语料库，本课题针对短问句的语义匹配算法进行了相关的研究，通过使用词向量来进行问句的表述，对比了传统的基于相似度的算法、基于卷积神经网络的算法以及基于注意力机制的卷积神经网络的算法的优劣性，并作出选择。其中本课题所改进的基于注意力机制的卷积神经网络算法既具有能够提取高层的抽象语言特征的优点，同时又针对一些有效的底层特征进行了自动选择，因此取得了优于其它几种方法的效果。基于上述的语义匹配模型以及传统的信息检索和语义分析的技术，本课题搭建了一个自动问答系统用于阿里巴巴公司内部特定领域的常用问题的自动回答。

本课题的原始实验语料主要来自于百度知道及阿里云客服，在对这部分语料进行处理之后将它们作为标准数据集进行模型的训练。通过对各个模型进行对比实验可以发现基于注意力机制的卷积神经网络模型取得了最好的效果，F1 值达到了 78.3%。本课题在应用阶段构建了一个针对电商领域的自动问答系统，使用容器服务对系统进行线上部署，系统返回的准确率达到了 84.7%。

关键词：自动问答系统；语义匹配；卷积神经网络；注意力机制；容器服务

Abstract

With the rapid development of information technology, users' access to network resources are constantly changing. From initial yellow pages to traditional search engines, and then to current intelligent question-answering robots directly providing answers, such changes are made thanks to the significant progress in natural language understanding (NLU) and information extraction (IE). Users are increasingly inclined to use a simpler way to obtain information, which also requires computers to have stronger language-understanding ability. In recent years, more and more technology companies and research institutions have started to develop intelligent question-answering robots, such as Siri, Apple's intelligent voice assistant and Xiaoice launched by the Microsoft Search Technology Center, and the automatic question-answering module is a critical module of intelligent question-answering robots. Therefore, the automatic question-answering field has become popular in artificial intelligence researches. Besides, various new machine-learning methods applied in this field have also raised the intelligence level of question-answering systems. The purpose of this research is to build an automatic question-answering system for frequently asked questions.

This thesis mainly includes the construction of open semantic-matching corpus, the design of semantic-matching algorithm, and the construction of an automatic question-answering system. In view of the fact that there is no open Chinese corpus in the field of semantic matching, and the most widely used MSRP corpus has only limited amount of data, this project constructed an open semantic-matching corpus that can be used in Chinese question-matching. Based on the corpus constructed, the project made some research on the algorithm for short-sentence semantic matching. By expressing questions through word embedding, this project compared the merits and demerits of the traditional similarity-based algorithm, the algorithm based on convolution neural network and the convolutional neural network model based on attention mechanism. Then a choice was made. The improved attention-based convolutional neural network algorithm has the advantage of extracting high-level abstract language features and automatic selection of some effective underlying features, which is superior to other methods' performance. Based on the above semantic matching model and the traditional techniques of information retrieval and semantic analysis, this project built an automatic question-answering system for the frequently asked questions of Alibaba's internal domain.

The original experimental corpus of this project is mainly from Baidu Knows and Alibaba Cloud customer service. After being processed, this part of the corpus became a standard data set for model training. The results of the experiments show that the

convolutional neural network model based on attention mechanism performed the best, with the F1 score reaching 78.3%. This project constructed an automatic question-answering system for e-commerce in the application stage. The docker service was used to deploy the system, and the accuracy of system return reached 84.7%.

Keywords: automatic question answering system, semantic matching, convolution neural network, attention mechanism, docker service

目 录

第1章 绪　论

1.1 课题来源

本课题来源于哈尔滨工业大学（深圳）智能计算研究中心与阿里巴巴云计算的自动问答合作项目，目标是使用深度学习为基础构建一个针对垂直领域的常见问题集（Frequent Asked Questions，FAQ）自动问答系统，本人主要负责项目中的算法设计以及部分系统模块的实现。

1.2 课题研究的目的及意义

随着计算机科学与技术的飞速进步以及互联网的高速发展，人们在网络上能够获取到的信息也在不断增加，各种电子媒介新闻、广告蜂拥而至，导致了一种信息过载的现象。

在互联网刚出现时，人们在访问自己需要的资源时首先需要获取到该资源的网络资源地址。因此，尽管此时的互联网上信息较少，普通用户也难以获取到有用信息。搜索引擎的出现极大的改善了这个问题，1994 年成立的雅虎公司推出了第一代搜索引擎，它将互联网上的常用资源搜集起来整合到一个搜索服务网站中，对所有网站进行人工编辑并按照资源类型分类。然而随着互联网的增长，人工编辑的方式逐渐无法满足信息过载的需求，第二代搜索引擎应运而生。1999 年，谷歌公司通过使用网络爬虫获取互联网资源，并利用新型的信息检索技术对所有资源进行高效的索引，推出了新一代谷歌搜索引擎。通过这种方法，用户有可能检索到链接到互联网中的任何有用资源，而搜索引擎公司的成本也大大降低。

近几年，随着移动终端的迅猛发展，移动互联网逐渐成为互联网中一个重大的组成部分。从服务形式上来看，信息检索仍是信息获取的主要方式，但由于移动终端的简易性以及用户的个性化需求，信息检索的内容已经逐渐发生了改变。在很多场景下，为了满足不同用户在不同领域的信息需求，我们要求搜索引擎反馈具有实时性、精确性和独立性。然而目前的搜索引擎容易获取词语级别的关联信息，但难以获取语义层面的回答。例如，面对用户搜索“去深圳机场怎么走”，搜索引擎可能难以理解用户的真正意图，返回一些问题的无关信息。同时，信息爆炸导致了搜索引擎经常会返回过多的无关信息，从而干扰到用户对有用信息进行选择。因此，为了让用户拥有更为舒适的人机交互体验，研究机构和科技公司开始使用问答（Question Answering，QA）技术来解决搜索引擎中存在的问题，问答系统也成为了目前信息技术的新一代研究热点[1]。

自动问答系统是目前人工智能领域的一个新型研究热点[2]，它是继信息检索技术之后的新一代信息获取系统。问答系统可以更精确的理解用户以自然语言形式描述的提问，并通过检索问答知识库返回一个简洁、精确的答案。从时间上来看，问答系统最早可以追溯到图灵测试，图灵在 1950 年提出了一种测试方法来评价机器是否拥有智能，这也成为了之后很长一段时间内评价人工智能的标准。随着技术的发展，问答系统本身的形式也在不断发生变化。早期由于技术的局限性，问答系统通常只针对某类特定领域，例如 BASEBALL[3]、STUDENT[4]、LUNAR[5]系统。这一时期的问答系统主要处理一些结构化数据，研究人员通过一些人工或者半人工的方法来构建领域知识库，耗费的资源较多。

随着互联网的飞速发展以及自然语言处理技术的提高，问答系统逐渐转入了开放领域，大量使用网络上的半结构化或者非结构化数据，通过自然语言处理技术建立大规模的开放知识库，解决多个领域的问题，例如英文问答系统 START、IBM 公司的知识百科竞赛冠军沃森系统[6]。

近年来，互联网的飞速发展导致了传统产业与互联网的联合，也就是人们常说的“互联网+”。这种改变也促使了一些需要大量人力的传统行业如服务业、咨询业等投入到计算机自动处理。许多公司和研究机构的人工智能项目都希望开发一个适用性较广的自动问答系统，从而帮助用户能更方便的获取信息以及节省公司内部的人力资源。比如，日本研究机构开发的针对大学入学考试的智能答题机器人[7]；京东公司在 2014 年推出的 JIMI 自动问答机器人，该机器人通过使用自然语言处理、机器学习等技术，能够对用户提供电子商务各个环节的服务；微软公司在 2014 年推出的个人智能助手微软小娜，主要作用是根据用户的个性化需求，为用户提供行程安排、问题回答等服务。阿里巴巴集团作为国内的大型科技公司，也比较迫切的希望构建一个可以针对集团内各个业务部门的自动问答系统，而本课题所属实验室在自动问答领域也有一定的经验与积累，因此与阿里巴巴集团展开合作，共同进行自动问答机器人项目的开发，本人主要负责语义匹配算法的设计与实现和问答系统的部分实现。

1.3 国内外相关技术研究现状

1.3.1 自动问答系统研究现状

人工智能最早的研究可以追溯到图灵测试，1950 年图灵测试的提出使得研究人员开始了自动问答领域的研究[8]。研究人员在人工智能、语义理解等相关领域取得的成果，也推动着自动问答研究的进步。

20 世纪中期，欧美的人工智能研究人员开始了自动问答系统的研究，由于数

据规模以及相关技术的局限性，此时的系统主要是面向特定领域的专家系统，可以认为是面向特定领域的自动问答系统的雏形，例如 BASEBALL 和 LUNAR 系统。BASEBALL 系统主要关注的是美国棒球联赛在一年内的相关问题，它的知识库储存一年内所有比赛的得分信息，来对用户提出的自然语言形式的棒球问题进行回答；LUNAR 是一个基于知识库的自然语言理解系统，主要服务于美国阿波罗登月计划，通过对阿波罗号月球探测器采集得到的地质数据进行收集和分析，来回答用户的问题。除此之外，1968 年美国 MIT 的博士生 D.Bobrow 开发了一个基于模式匹配的自然语言理解系统 STUDENT。该系统基于一些特定的启发式信息及模式匹配规则，能求解简单的英文表示的数学问题。这一时期的问答系统主要针对的是特定领域，而且需要一个由领域专家精心定制的知识库或者知识规则作为基础。

进入 20 世纪 90 年代之后，伴随着互联网的发展以及人工智能技术的进步，问答系统的性质也逐步转向了面向开放领域、基于互联网非结构化文本。例如英文的基于英特网的问答系统 Ask、START，这种系统主要包括以下几个处理流程：知识库构建、问题分析、文档检索、答案生成，通过对互联网信息的自动采集与整理，它能够回答科技、文化等多个方面的问题。1999 年，文本检索会议（Text Retrieval Conference）引入了问答系统评测专项（Question Answering Track），极大的推动了整个计算机领域在问答系统上的研究发展[9]。

随后，网络上出现了常见问题数据，特别是 2005 年以来出现的大量的在线互动问答社区系统（Community based Question and Answering Service）为研究者提供了大量的非领域限定的问题答案对数据，问答系统真正进入了开放式、基于问答对的时期[10]。

问答系统发展至今，已经成为了自然语言处理分支下的一个比较成熟的领域。从用户问题所属的领域范围来看，问答系统可以划分为面向开放领域的问答系统、面向限定领域的问答系统以及面向常用问题集的问答系统；从答案的生成方式来看，可以分为检索式（指答案是从知识库中的文本段落中抽取出来的）问答系统和生成式（指答案是通过一定的规则或者编码解码产生的）问答系统两大类。目前的通用自动问答系统主要包括问句理解、信息检索、答案生成、知识库构建几个模块，一个通用的整体框架如图 1-1 所示。

（1）问句理解相关技术

问句理解部分主要是负责分析和理解用户提出的问题，产生一些候选查询项从而协助后续的各个模块。就这一部分来说，理解用户意图是关键，也是问答系统区别于传统信息检索引擎的核心，它需要将用户表述的自然语言形式的信息转化为机器可识别的形式，传入到系统后续的模块中进行处理。在研究中通常把问

句类别和问句内容作为问句语义的表示，因此这部分一般有问句分类、问句主题提取两个主要的研究方向。

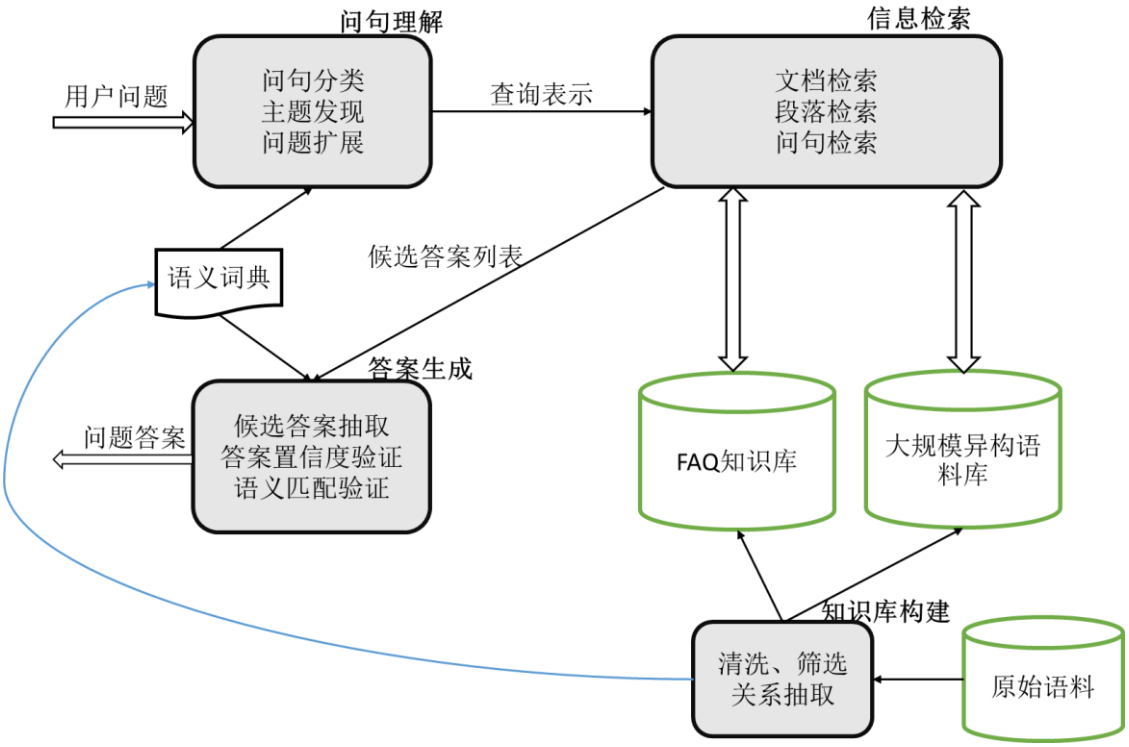

图 1-1 通用问答系统整体框架

问句分类是指将问题划分到具体的类别中，在之后的处理过程中根据不同的类别采取不同的策略。从学习理论上来说，问题分类的实质是缩小用户问题的解空间规模，以便高效准确地得到答案。研究人员通常采用模式匹配与机器学习结合的方法来进行问句分类，模式匹配是指人为设置的一些规则，如果问题落入到这一规则中，则认为问题属于某一类。而机器学习方法则是根据研究人员提取的问句特征集合，在训练数据上得到一个分类器[11,12]，然后用该分类器对新的问句进行分类。

问句主题提取主要目的是获取问句的主题焦点，主题是代表用户问题所感兴趣的领域对象，焦点则是用户问句的核心内容，获得的主题焦点进行组合后作为候选查询词传入到信息检索模块中。通常使用句法分析的技术来获取问句的中心词，然后使用词本身和该词的修饰语作为查询词，Cui 等人[13]提出一种基于外部搜索引擎返回结果的方式来选取主题词组，通过搜索引擎返回结果中各种词之间的点互信息来发现问题的主题。

（2）信息检索相关技术

根据问句理解模块提供的查询候选词，信息检索模块主要负责从问答知识库

中检索到问句相关的段落信息并对它们进行粗略的排序，缩小解空间的范围，然后将候选答案段落传输给系统中的下一模块。对于不同类型的问答系统，信息检索模块的结构也不尽相同。对于基于开放式的问答系统，信息检索通常要做的是文档检索和段落检索两个步骤；对于基于问句答案对的问答系统，信息检索通常需要找到与原始问句语义最相近的问句列表。文档检索常用方法包括向量空间模型、布尔模型、语言模型、概率模型等，段落检索常用方法包括 MultiText 算法[14]、IBM 的算法[15]和 SiteQ 算法[16]。问句检索的主要问题是如何衡量自然语言表述的用户问句和知识库问句之间的语义相似度。与传统的自然语言处理研究对象不同，问句检索主要关注的对象是句子或者短文本片段，这也就使得一些传统的语义相关性计算的方法变得不再适用。近几年研究人员将统计机器翻译的思想引入到问句检索领域，将两个不同形式的问题匹配看做是机器翻译中的翻译过程，基于这一思想，Riezler[17]等人提出了一个有效的模型来获取问句之间的语义相似度。总的来说，基于统计机器翻译的方法在扩展问句表述和计算问句语义相关性方面取得了一定的成果，效果比传统的向量空间模型，语言模型都要好。

（3）答案生成相关技术

基于信息检索模块的相关信息，答案生成模块需要从相关文本中生成候选答案集合，然后从中提取出正确答案。候选答案抽取主要负责从相关文本片段中对答案进行压缩提纯，生成一个简洁明确的答案。答案提取是将问题与现有的候选答案进行语义上的进一步验证，从而保证返回给用户最相关的结果。从句子表层特征来看，可以用一些答案周围文本片段的语法、语义特征来衡量答案的置信度。目前使用比较广泛的是基于统计机器学习的答案置信度衡量方法，研究人员通过定义一系列的词法、句法、语义等其它相关特征来对问题和候选答案之间的相关性进行表述，然后使用分类器的分类置信度表示候选答案的置信度。

在过去，由于自然语言处理技术的局限性，深层的分析模型一般不能达到工程使用的效果，问答系统构建过程中用到的机器学习模型通常属于浅层模型。例如问句分类中用到的支持向量机（Support Vector Machine，SVM）分类模型，问句分析中命名实体识别用到的条件随机场（Conditional Random Fields，CRFs）、隐马尔科夫（Hidden Markov Model，HMM）序列标注模型等。这些浅层的模型在特征选择阶段往往需要大量的人工介入，在不同领域的可移植性也较弱。

近几年，深度神经网络在图像处理、语音识别领域取得了重要的进展，显示出了极为优越的表示学习能力。与此同时，研究人员试图在自然语言处理领域引入深度学习模型，目前也已经取得了一些阶段性的成果。例如，Bengio[18]等设计的神经网络语言模型（Neural Network Language Mode，NNLM）得到了一种名为

词向量（Word Embedding）的新型词语向量表示，这种词向量具有低维度、稠密、可直接计算等特点，并且能比较完善地表示词义信息以及词法信息。同时，循环神经网络（Recurrent Neural Network，RNN）、卷积神经网络（Convolution Neural Network，CNN）、递归神经网络（Recursive Neural Network，RNN）也被应用到 NLP 中的各个任务中。在问答领域相关的任务中，Kim 提出了一个新型的浅层卷积神经网络[19]，通过词级别的卷积以及一系列池化的组合对句子进行分类，在多个数据集上取得了历史最高的效果，答案选择[20]和答案生成[21]方面也同样有一些新型的深度学习算法出现，并且获得了较好的结果。

1.3.2 卷积神经网络研究现状

卷积神经网络是近年来广泛用于图像处理、模式识别等领域的一种特征提取分类方法。卷积神经网络的历史最早要追溯到 1962 年，研究人员根据动物的感知信息提出了感受野的概念。1998 年 LeCun 等人提出的用于手写数字识别的具有 7 层卷积神经网络结构的 LeNet-5 系统在工业界取得了较大的成果，可以认为是较早的使用卷积神经网络的工业应用。如今，随着可训练数据规模的扩大和计算机计算性能的增长，卷积神经网络在很多任务上都取得了不错的效果，逐渐成为机器学习领域新的研究热点。相比于全连接神经网络来说，卷积神经网络由于引入了感受野和权值共享的概念，减少了网络训练所需要的参数的数量，从而加快了运算速度。对于图像处理以及模式识别任务来说，卷积神经网络直接以原始图像（或者作简单处理）作为输入，无需复杂的特征构建过程，缩短了算法模型开发的周期，也增强了模型的可移植性。

从卷积神经网络结构上来说，卷积神经网络的核心由卷积和池化（Pooling）两个部分构成。自然图像本身具有一些固有特性，可以认为图像的某一局部区域的统计特性与其他部分是相同的，即在图像某一部分学到的参数特征也可以用在图像的其他部分上，基于这一思想，引入了卷积的概念。卷积层的前后两层采用非全连接的方式连接，通过卷积运算对前一层进行局部特征提取，随着卷积窗口的移动逐渐生成当前所有局部信息。在卷积操作之中引入激活函数（例如 sigmoid、tanh、ReLU）进行激活，由于采用权值共享的方式，网络的复杂度也大大降低。而在卷积层中引入多个卷积核也使得网络可以获得输入层不同层次的局部特征进行组合，增强了网络的表示能力。池化通常连接在卷积层之后，主要作用是对卷积层得到的特征进行一种局部信息的归一化，例如，人们可以取矩阵中某个窗口内的平均值（或最大值）。池化后的统计特征不仅具有更低的维度，同时对网络的训练也有一定的帮助（防止过拟合）。

时至今日，卷积神经网络已经在图像处理、模式识别等领域取得了重大的进展。从 1999 年 LeCun 等人使用卷积神经网络进行手写数字识别起，卷积神经网络在图像识别方面不断取得新的进展。2012 年，Alex 在图像分类领域提出的 AlexNet 网络奠定了卷积神经网络在计算机视觉领域的地位[22]。随着深度学习技术的发展，其他技术领域也逐渐开始引入卷积神经网络进行研究。在自然语言处理方面，早在 2006 年 Bengio 等人就提出了一个神经概率语言模型来建立新型语言模型，而这之后的工作都主要集中在循环神经网络和递归神经网络上，例如，Mikolov 等人依照自然语言的词序特点使用循环神经网络对语句进行建模，在句法分析，命名实体识别等领域都取得了不错的效果[23,24]。2013 年以来，研究人员开始把卷积神经网络引入到自然语言处理的任务中。在自然语言处理中通常使用词向量矩阵作为网络的输入，在模型改进方面，Kalchbrenner 提出了使用一个卷积神经网络对每个句子进行建模[25]，Kim 使用一个浅层的词级别的卷积神经网络来进行问句分类，Hu 提出一个并行的卷积神经网络架构通过分别对两个句子进行建模来完成语义匹配的任务[26]。今年的人工智能热点、战胜围棋人类冠军的 AlphaGO 系统同样在估值网络的训练过程中引入了深度卷积网络作为训练模型[27]。

1.3.3 语义匹配算法研究现状

语义匹配算法的研究在自然语言处理及其相关领域已经持续了很多年。语义匹配算法的最早应用是在信息检索中，研究人员通过设计语义匹配算法来衡量检索文档与用户查询之间的相关性，并对检索文档进行排序。而在相关反馈，文本分类，词义消歧以及最近的文摘抽取，文本摘要等领域，文本相似度也起到了非常重要的作用。2006 年，微软公司的研究人员使用多个新闻媒体作为素材进行平行语料的抽取，使用无监督的方法构建了一套英文的基于新闻语料的语义匹配语料库 MSRP[28]，之后的语义匹配算法研究人员大多以此语料库来进行使用。

在早期的语义匹配研究工作中，研究人员大多使用一些比较简单的、无监督的方法进行文本相似度的计算，例如潜在语义分析（Latent Semantic Analysis，LSA）、点互信息熵（Pointwise Mutual Information，PMI），或者是引入一些人为的知识信息，例如 WordNet 或者其它语义信息。随着支持向量机等机器学习算法的发展，研究人员开始将语义匹配任务作为一个分类问题进行处理，提取句子的表层特征进行分类。2006 年以来，一些研究人员把机器翻译的思想引入到语义匹配中，他们将两个句子的匹配视为不同语言的翻译问题，引入多种机器翻译评估的自动度量标准（BLEU、NIST、WER 等）作为问句对的特征，然后用训练语料的特征训练一个分类器，这种方法比起传统的无监督方法在 F 值上有了比较显著的提升。

2011 年，Socher 提出了一种基于动态池化和树结构自动编码的神经网络架构来对句子对整体进行建模，这种方法既获取了句子间词级别的关联信息，句法树的存在又使得短语级别的信息也被抽取出来，在当时取得了历史最好的效果[29]。2014 年，Hu 把卷积神经网络架构应用到了语义匹配上，通过一个并行的改进的卷积神经网络，两个句子的特征分别被抽取出来，最后输入到一个分类器中进行分类。这之后，He 等人对其中的卷积结构进行改进，加入了一个新的相似度层并在卷积时引入了多视角（Multi-Perspective）的概念，在该任务上取得了非常好的效果[30]。

1.4 本文的主要研究内容

本研究的主要方向是自动问答、语义匹配，目的是使用目前流行的深度学习方法构建一个移植性较强的自动问答系统。针对本人的研究目标，本课题研究内容应该是设计一个适用性较强的语义匹配算法以及结合信息检索以及语义匹配算法，构建一个自动问答系统。综上所述，本课题主要针对以下几个方面展开研究。

（1）构建一个适用于语义匹配项目的中文数据集。语义匹配作为一个 NLP 领域传统的项目，本身已经有了一些比较优秀的语料库，例如微软发布的 The Microsoft Research Paraphrase（MSRP）corpus 以及另一个使用比较广泛的 User Language Paraphrase corpus。但在中文语义匹配领域，至今还没有一个被大家广泛认可的转述识别语料库。因此构造一个开放性中文问句转述识别语料库是算法设计的前提，这部分也是课题所研究的重点之一。

（2）语义匹配算法的设计。语义匹配算法本身就是问答系统的核心组成部分，因此在语料库构建完成之后，我们需要考虑设计合理的语义匹配算法进行实验。考虑到目前复述识别领域已有的研究成果，本课题构造了有监督的和无监督的两大类方法进行对比实验。

（3）问答系统的实现。在核心的算法模块设计完成之后需要考虑系统的实现工作，这部分工作主要包括问答系统本身的几个关键部分，问句分析、信息检索及语义匹配、答案生成。在实际的应用场景中，如何将设计好的的语义匹配算法应用到系统中也是一大关键。

1.5 本文的章节结构

第 1 章是绪论，首先介绍课题来源，接着阐述课题的目的及意义，然后对课题所涉及的相关领域及技术进行分析。

第 2 章是开放式的语义匹配语料集的构建，首先确定原始语料的来源，接着使用多种方式对初始语料进行筛选，最后设计实验方案对语料进行评估。

第 3 章是语义匹配算法的设计，首先简单介绍本课题所使用相关技术的原理，接着逐个介绍课题在语义匹配方面使用到的相关算法，包括有监督和无监督，浅层模型和深层语义模型，接着使用构造的语料集设计相关的对比实验，确定最后的匹配策略。

第 4 章是自动问答系统的构建过程。这部分将传统的自然语言处理流程与算法核心模块所获得的算法模型相结合，实现一个可用于工程的自动问答系统，并且尝试在具体领域进行问答系统的搭建，探索一些领域相关的优化方法。

第 5 章是实验结果及分析，主要分为语料集构造相关实验和语义匹配算法相关实验两大部分，通过对不同的方法进行对比，选取最优的策略。

第2章 开放式语义匹配语料集的构建

在有监督学习方法的研究领域，语料集是算法改进创新的基础。由于本课题希望设计一个适用性强的语义匹配算法用于问答系统中，因此首先需要构建一个开放性的语料库。

2.1 语料集构建的目的及意义

语义匹配作为一个 NLP 领域传统的任务，本身已经有了一些比较优秀的语料库，比如 MSRP、PAN[31]、PPDB[32]但是这些语料库本身存在着一些缺点。MSRP 语料是基于新闻语料构造的一个针对长文本陈述句的复述识别数据集，数据集本身只有 5799 对句子，而目前的深度学习算法由于模型比较复杂，参数规模较大，通常需要大量的数据进行训练。而 PAN 语料本身是一个抄袭检测的语料库，虽然规模较大，但其中包含比较多的噪声信息，并不适用于我们的问答任务。最重要的问题是，本课题的问答系统主要是针对中文文本的自动问答系统，语义匹配算法在其中所起到的作用是衡量两个问句之间的相关性。基于以上原因，本课题构造了一个大规模的中文问句复述识别数据集，用以对所研究的语义匹配算法进行训练。

2.2 原始语料获取

2.2.1 初始文本信息获取

由于本课题中语义匹配算法最后的应用场景是中短型问句的匹配，因此在语料集的构建上，本课题需要构建一个中短型问句的语料集。考虑到目前在线互动问答社区（例如 Yahoo！ Answers，百度知道）本身已经发展的比较成熟，而且这些问答社区的问句具有多样性、开放性的特点，其中的语料类型与本课题所研究的任务非常相似，我们希望直接获取这些问答社区的数据作为我们的原始语料集。因此，本课题使用阿里巴巴内部提供的搜索接口采取多种策略进行搜索，以获得原始语料。搜索策略如下所示：

（1）使用单关键词进行作为搜索引擎的种子。考虑到需要在搜索引擎中进行召回，我们首先需要指定搜索的查询词，采用单个词作为查询词进行搜索召回。考虑到构造的语料主要面向于开放式的复述识别问答系统，检索得到的语料应该覆盖到尽量广泛的领域，因此本课题使用搜狗细胞顶层词库的所有词汇数据作为种子词进行检索，得到检索结果进行存储。

（2）使用多关键字组合作为搜索引擎的种子。与策略（1）相类似，在这种策略中，首先使用搜狗细胞词库的顶层词汇作为种子进行检索，得到一个顶层知识库。在获得顶层知识库之后，对属于每一个种子词的问句做分词处理，根据词的词性及 tf-idf 值进行关键词抽取，获得关键词对。下一步对关键词的出现频率做统计，取频度超过阈值的关键词对作为种子查询词。最后使用新生成的种子查询词进行搜索，得到检索结果进行存储。

（3）使用整句作为搜索引擎的种子。在这种策略中，首先通过阿里巴巴的内部搜索渠道获得一部分开放领域的短问句作为初始集，接着对初始集进行筛选，去除一些不规范的句子及重复问句，使用筛选过后的句子列表作为搜索引擎的种子进行检索，得到检索结果进行存储。

使用策略（1），选取初始集后获得的结果如图 2-1 所示（以词 理财为例）：

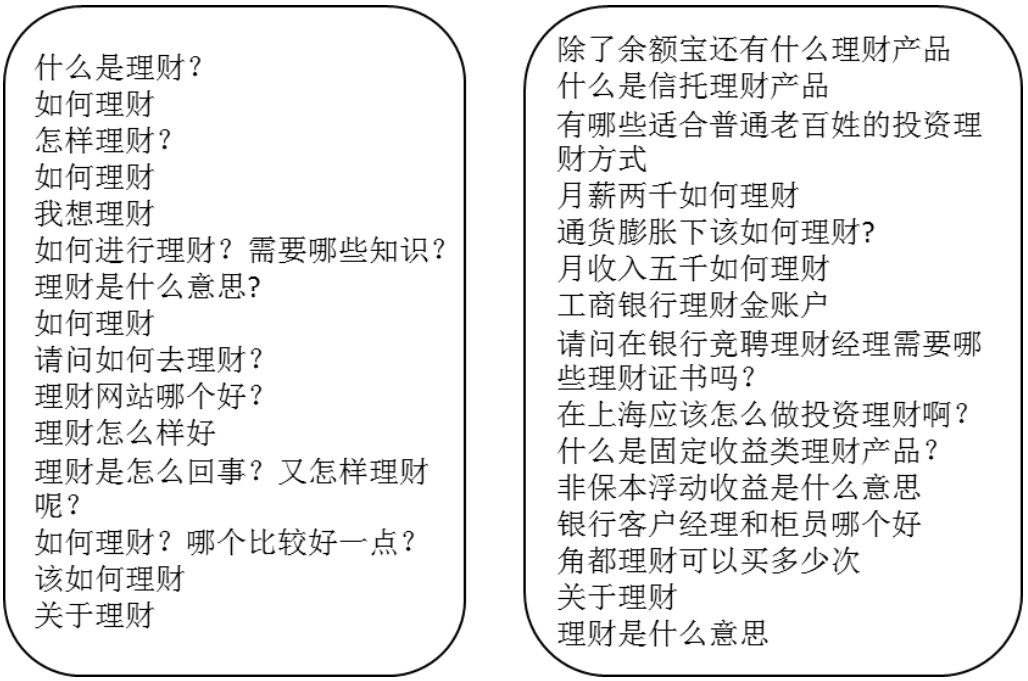

图 2-1 “理财”召回结果示例

图 2-1 中左边文本框显示的是检索返回的前十五个结果，图 2-1 中右边文本框显示的是检索返回的后十五个结果。很容易发现前十五个结果的语义相对比较集中，而后十五个结果的语义比较发散，大多数句子之间表示的是完全不同的意思。对所有返回结果进行分析可以发现在使用单个词作为种子得到的检索结果中，虽然大部分句子之间的意思都比较相关，但是如果直接把它们随机进行组合作为正例的精度会比较低，例如“如何理财”、“理财网站哪个好？”、“理财是什么意思”这三句话分别代表了不同的意思，因此我们需要根据句子语义层面的信息，

采取一些更深入的筛选方法来获得精度比较高的正例。参考 MSRP 语料集的自动构造方式，在获得检索结果后，本课题采取了一些模式匹配以及机器学习的方法来自动构造正例。

2.2.2 问句对抽取

在构造正例时我们需要将表达同一种意思的句子划分到同一类中，这也就引出了聚类算法。聚类是一种无监督学习过程，与有监督学习方法不同，这种机器学习方法通常不会预先准备好已标注的训练示例，而是在聚类学习的过程中自动对样本进行类别划分。从某种意义上来说，聚类是一种通过建模简化数据分析的方法，通过聚类发掘到类内部的共同特征以及类间的区别特征，从而能在其他的算法中各全面的对数据特征进行建模。从聚类的目的来看，聚类将数据分到了不同的类别之中，同一类别中的对象具有较高的相似程度，而不同类别之间的对象则具有比较大的相异性。比较常用的聚类方法包括 k-均值（K-means）、k-中心点聚类等算法。

而从语法结构上来说，一般来说，两个具有相同含义的句子的语法主体部分是类似的，例如“我想要退回这两件衣服”和“麻烦一下两件衣服一起退货”。可以发现，简单地来看两个句子从整体句式和词语构成上都有一些区别，但对分别对这两个句子的词性进行分析，可以发现它们句子中的名词，动词，代词成分都具有比较高的一致性。对搜索获得的问句分析后可以发现，人们日常的自然语言表述中，经常有很多语气词，感叹词等等，这些词往往对语言本身的意义没有太大的影响，但是却会对计算机的语义分析过程产生很大的障碍，而这些词在人为的设置一些模式匹配规则的过程中都可以过滤掉，大大减少了计算机处理过程中的噪声。因此，针对词性、句法结构设计的模式匹配规则也可以帮助我们进行正例的自动构造。综上，本课题在正例自动构造上提出了三种筛选策略。

（1）k-means 聚类。k-means 聚类是一种常见的基于距离的快速聚类方法，采用距离作为相似度的衡量标准，认为在向量空间中两个样本距离越近则具有越高的相似度。在本课题中数据样本是自然语言表述的问句，无法直接衡量它们之间的距离，因而需要将中文自然语言表述转化为向量编码，在这里采用了 2013 年 Mikolov 提出的 word2vec 算法来对词语进行向量编码[33,34]，具体原理将在下一章进行介绍。对每个种子获得的问句列表，分别对它们进行分词操作，将这些词语使用 word2vec 转化为词向量，使用句子内所有词向量求和取平均的结果来表示单个句子。最后分别对每个种子的问句列表进行聚类操作，类别数为 3。

（2）关键词聚类。关键词聚类的基本思想与前者类似，但是这种方法直接使

用词的共现来评估两个句子之间的相似度。因此，对所有种子获得的问句列表，直接将它们视为一个整体来进行关键词聚类，聚类的策略比较简单，具有相同关键词的句子归入到同一类中。在关键词的提取使用了两种方法，其一是使用 tf-idf 加词性规则。首先对句子的每个词计算 tf-idf 值并进行分析，设置一个合适的阈值用来筛选关键词，接着对句子的词性构成进行研究，分析哪些词性对句子语义的影响较弱。在确定了阈值和特定的词性规则之后，对每个句子进行分词，对每个词的 tf-idf 值和词性进行分析，高于阈值及符合词性规则的则作为句子的关键词。另一种关键词提取方法是 TextRank[35]，这种算法基于谷歌早期提出的 PageRank 的思想，可以用来提取关键词和生成文本摘要。

（3）模式匹配。传统意义上的模式匹配算法其实是对字符串进行的一种基本运算，即给定一种子串模式，要求在字符串中找出与该模式所匹配的所有子串，子串模式的表示即可以是一个实际字符串也可以用一种统一的模式来代表，在高级编程语言里通常用正则表达式来实现。而在本课题的这个筛选任务中，我们使用的是一个广义的模式匹配的概念，即我们设计一些特定的语法、语义规则对两个问句进行匹配，符合规则的问句对我们认为是正例。例如，名词、动词的相似度必须在阈值之上，代词需要指代相同的含义但允许某个句子缺失。相似度的衡量方法可以采用一个预训练好的 word2vec 模型，也可以引入 HowNet 进行语义相关性的度量。

在策略（1）和（2）中我们得到了很多不同的类别，可以发现在一个类中的句子通常表示同一种意思，而在不同类中的句子通常表示不同意义。因此，我们随机抽取相同类内的句子组成候选正例问句对，选取分别属于不同类别的句子组成候选负例问句对。而在策略（3）中，通过模型匹配的方法我们可以直接获得候选正例问句对，在不同种子查询之间的问句列表中进行抽取可以得到候选负例问句对。数据集整体构造流程如图 2-2 所示，使用“理财”作为初始种子，选取多关键词检索和 k-means 聚类策略构建过滤对正例的具体流程如图 2-3 所示。

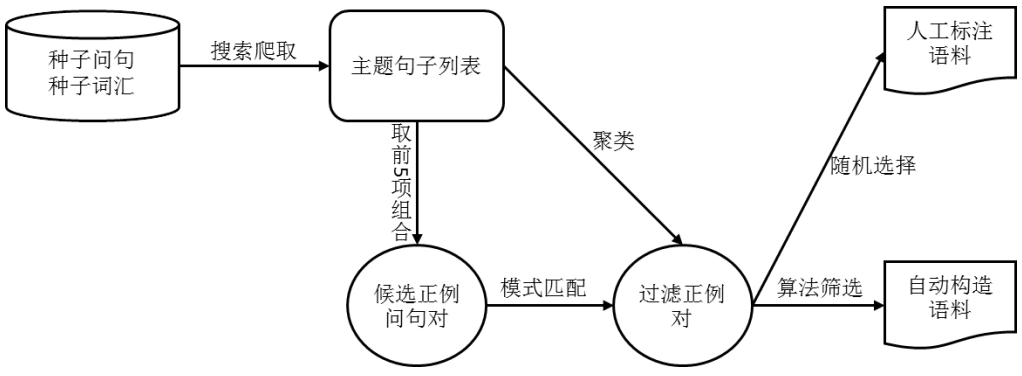

图 2-2 数据集构造流程图

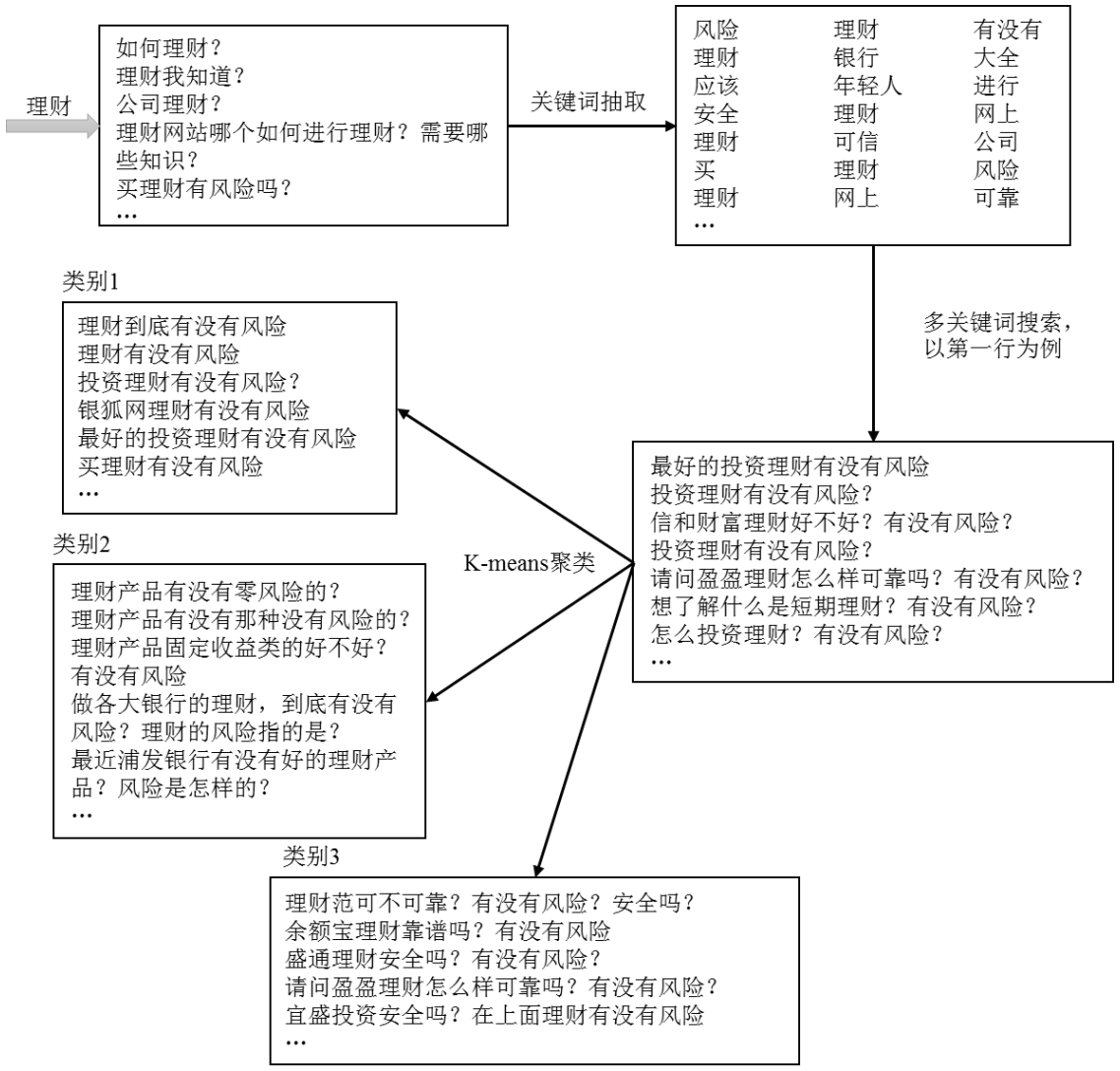

图 2-3 “理财”构造数据流程示例

2.3 语料集的筛选及标注

2.3.1 人工标注

在使用自动的方法获得了原始语料之后，我们决定随机抽取出其中的一部分来进行人工的标注工作。人工标注既能验证之前的自动方法是否有效，又能为语义匹配算法模型的建立提供数据基础，因此，我们从所有候选正例问句对中抽取出 240660 个条目进行人工标注工作，每条问句由三人同时进行标注，问句之间相互为复述标为 1，不为复述标为 0，不确定标为 N。对于每个条目来说，标注完成后都会具有三个类标，然后我们使用投票的策略来决定每个条目的最终类标，也就是如果某个类标出现 2 次或者 2 次以上，并且 1 和 0 不会同时出现，则将出现多次的这个类标作为数据的最终类标；如果不符合上述条件，则将数据标记为 N。

数据标注举例如表 2-1 所示。

表 2-1 数据集人工标注示例

问句对	TAG1	TAG2	TAG3	类标
S1：电脑用什么软件下载游戏好 S2：用电脑下载游戏什么软件好	1	1	1	1
S1：梦幻西游股票怎么取钱 S2：梦幻西游怎么把买股票的钱取出	1	N	1	1
S1：埃及是哪个洲首都是 S2：埃及的首都是哪	0	1	0	N
S1：马油怎么用效果好 S2：马油效果怎么样	0	N	0	0

可以看到，通常如果在某一条条目中同时出现了 1 和 0 两种类标，那么意味着这个两个句子之间的联系非常模糊，标注人员很难客观的判断两个句子之间的关系，因此这种情况下将数据条目标为 N 是合理的。三个标注人员同时标为 1 或者两个标注人员标为 1，一个标注人员标为 N 则表示有比较大的把握确定两个句子表示相同的语义。最后对标注完成之后的所有数据条目进行统计，结果如表 2-2 所示。

表 2-2 人工标注结果

正例	负例	未知	总计
138110	102550	9334	249994

2.3.2 自动构造

考虑到深度学习模型的训练通常需要大量的数据，因此我们还希望自动地构造一些精度较高的训练数据来进行训练。在这里本课题同时使用了一种无监督的方法叫做词移动距离[36]（Word Mover Distance，WMD）对剩余的候选正例问句对进行进一步的过滤，通过引入这种无监督的算法来构造大量的数据集并使用这些数据对模型进行训练从而学到更广泛的模式。

与传统的 TF-IDF 以及 LDA 算法的作用相似，WMD 是一种新的基于词距离的文档距离度量算法，这个算法的基础是利用大规模的无监督语料学到词语的语义信息并用词向量进行表示，词向量可以学到类比信息并且已经被证实了词向量之间简单的算术运算是合理的，因而使用词向量的余弦相似度来度量两个词之间的距离也是合理的。WMD 的思想来源于计算机科学中地球移动距离（Earth Mover

Distance，EMD）的概念，EMD 表示在某个区域中两个概率分布距离的度量，最早被用于图像处理及语音识别领域。而在 WMD 中，研究人员使用词的分布来替代概率分布，使用词向量的技术得到两个词之间的距离，从而得到了两个文档之间的距离，计算方法如式（2-1）、（2-2）、（2-3）、（2-4）所示。

$$EMD(\mathrm{P},\mathrm{Q}) = \min_{\{f_{ij}\}} \frac{\sum_{i,j} f_{ij} d_{ij}}{\sum_{i,j} f_{ij}} \tag{2-1}$$

$$\text{s.t}: \quad \sum_{j} f_{ij} \le P_i \qquad \sum_{i} f_{ij} \le Q_j \tag{2-2}$$

$$\sum_{i,j} f_{ij} = \min(\sum_{i} P_i , \sum_{j} Q_j) \tag{2-3}$$

$$f_{ij} \ge 0 \tag{2-4}$$

式中 f_{ij}——句子 1 中第 i 个词转化到句子 2 中第 j 个词的转化权重；

d_{ij}——句子 1 中第 i 个词语与句子 2 中第 j 个词语之间的距离；

P_i——句子 1 中第 i 个词语的权重；

Q_j——句子 2 中第 j 个词语的权重。

其中式（2-1）求解了两个句子分布之间的距离，式（2-2）的不等式约束了转化权重不得大于该词在句子中的权重，对两个句子转化示例如图 2-4。

对所有的候选正例问句对，使用 WMD 算法求得问句对之间的距离，选取距离接近 0 的问句对作为最终自动构建的正例。而对于所有的候选负例问句对，使用 WMD 算法求得问句对之间的距离，选取距离值接近 1 的问句对作为最终自动构建的负例。

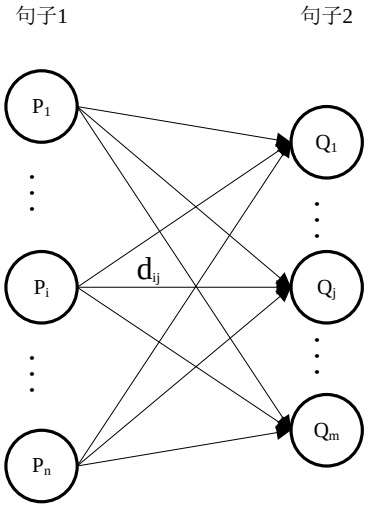

图 2-4 两个句子间的 EMD 算法

2.4 语料集评估

本课题在这一阶段的目标是构建一个大规模的中文问句复述识别语料库，语料库由四部分构成，人工标注训练集，自动构造训练集，验证集，测试集，其中人工标注训练集、验证集和测试集都是由人工标注得到的，验证集主要用于机器学习方法超参数的调整和停止条件的设置，具体数目统计如表 2-3。

表 2-3 数据集构成分析

数据	总计	正例	负例
人工标注	240660	138110	102550
自动标注	500000	200000	300000
验证集	8802	4402	4400
测试集	12500	6250	6250

对数据集进行分析，在数据规模方面如上表所示，通过人工标注得到了 240660 条训练集，这一部分数据正负样本比为 1.35:1，自动标注得到了 500000 条训练集，正负样本比为 2:3。从训练集角度来看，不管是对于传统的机器学习方法或者是深度学习方法来说，这样的数据规模都可以比较好的体现出模型的性能。测试集总共包括 12500 条问句对，其中正负样本比为 1:1。

从数据质量上来看，在人工标注的数据中有 73.1%被标注成了正例，这也就意味着原始语料中的候选正例精度为 73.1%。接着我们使用 WMD 算法对这部分候选正例进行了进一步的筛选工作作为人工构造的正例，在其中随机抽样 100 条进行检查，我们发现超过 80%的样本人工标注都为正例，也就意味着最后自动构造的正例精度超过 80%，我们认为这种精度的样本会对我们的模型训练产生一定的帮助，这种观点也会在之后的章节中得到验证。

从数据长度来看，词的长度主要集中在 8 到 18 之间，而字的长度主要集中在 9 到 22 之间。字数为 10 的句子在所有长度中占比最高，词数为 10 的句子在所有长度中占比最高。将句子分成字、词两种形式做统计的原因是汉语语言的结构与其他以空格划分的语言不同，中文的字和词本身都具有一部分的语义信息，虽然词语具有比较完整的语义信息，但是在我们进行自动分词的过程中通常也会有引入一些分词错误，因此在将语言信息输入到神经网络中的时候，我们通常会考虑分别使用单字的和分词的句子进行处理，然后分别训练一个深度学习模型，最后做出对比之后才会考虑选择某一种来进行实际系统所使用模型的训练，对整个数据集的句子长度统计如图 2-5 所示。

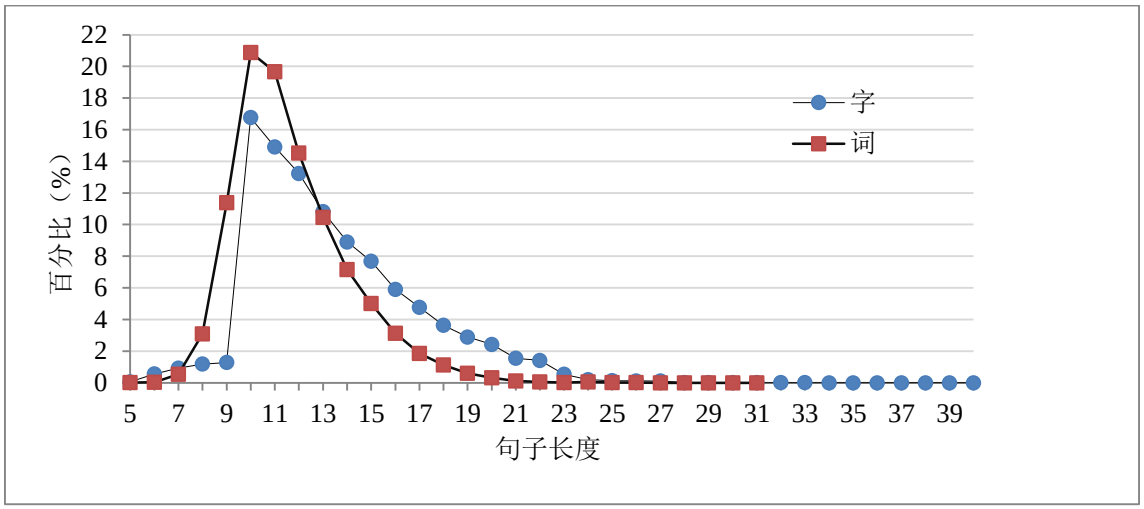

图 2-5 数据集句子长度分布

2.5 本章小结

本章首先介绍了构建一个大规模的问句复述识别语料集的目的及意义，然后依照语料集的构建流程分别对如何爬取原始的文本信息，如何对文本信息进行初步的筛选并组合成候选问句对，如何使用候选问句对自动地或者人工地构建一个语料库进行了详细的介绍，并在最后一小节里对本课题所构建的数据集的质量进行了详细的分析。

第3章 语义匹配算法设计

目前语义匹配复述识别算法主要包括两大类，一类是传统方法，包括使用无监督学习的 tf-idf、LSA、WMD 和使用机器学习方法的 MTMETRICS[37]、TF-KLD[38]等，一类是基于深度学习的方法，例如 SHPNM、ARC-II、Multi-Perspective CNN 等。本课题采用了多种语义匹配算法进行实验，最后对各个方法进行对比，选取合适的算法模型作为自动问答系统的核心算法模块。

3.1 词向量转化

文本的表征学习一直以来都是学习问题的重点之一。在早期，信息检索领域的研究人员使用稀疏词袋来表示一段自然语言文本，但这种方法往往会占用大量的存储空间，因此人们提出了潜在语义分析的方法来对稀疏的文本向量进行降维，这种方法学到了一种低维度、稠密的文本向量表示，但是这种方法只获取了词语之间最简单的共现关系。而深度学习作为一种优秀的表示学习方法，通过多层的特征学习并在每一层对非线性节点进行简单的组合来获取到更高层次的、精致的、抽象的特征信息，利用这些特征信息我们可以高效地进行分类、预测等工作[39]。2006 年 Bengio 使用神经网络建立语言模型，将深度学习引入到了自然语言处理领域，基于这一工作，Mikolov 发表了一系列相关的改进论文，并在 2013 年开源了 word2vec 算法模型进行向量空间中词语表示的学习。本课题中使用 word2vec 算法作为词向量的学习方法，将在以下两个小节中简单介绍 word2vec 的原理和训练方法。

3.1.1 词向量原理简述

词向量是由 Hinton、Bengio 等人提出的一种具有语义信息的文本特征表示，2006 年 Bengio 首先使用了一个三层的神经网络对词向量进行学习。2011 年以来，Mikolov 投入到文本表征学习中并做了一系列工作。2013 年，他提出了 Skip-Gram 和 Continous-Gram 两种语言模型对词的表示进行建模，Skip-Gram 使用窗口中心的词语对窗口内所有词语进行预测，而 Continous-Gram 则使用窗口内的词语对窗口中心的词语进行预测，这两种模型在其他方面的结构基本相同，因此下文中均采用 Skip-Gram 作为举例。比起 Bengio 提出的神经网络语言模型，Skip-Gram 在中间层中省略了一个非线性的隐藏层，从而大大减少了模型训练所需时间，相比传统的 LSA 方法，Skip-Gram 更大程度上地保留了词语之间的线性规则，模型的基本结构如图 3-1 所示。

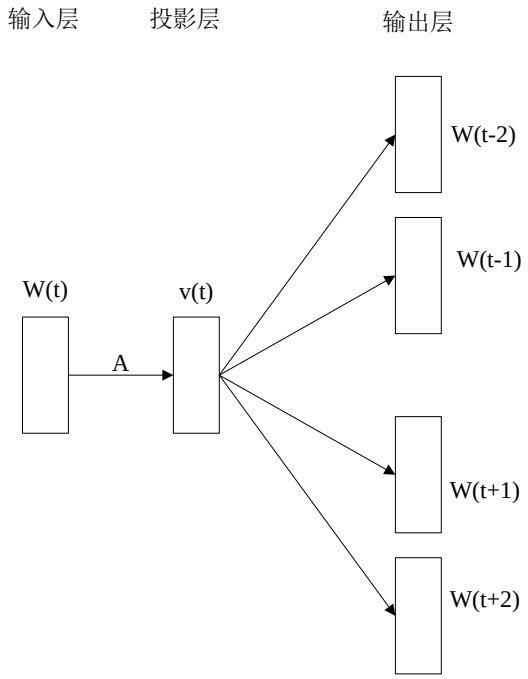

图 3-1 word2vec 训练算法

对于 one-hot 表示的词语输入$\omega(t)$，先经过一个投影矩阵进行投影得到中间层的向量表示$v(t)$，接着对向量表示$v(t)$进行 softmax 分类，分类目标是最大化$\mathrm{p}(\omega(t)|\omega(t+i))$的概率。使用传统的 softmax 方法进行分类，则 softmax 函数定义的分类概率如式（3-1）所示。

$$\mathrm{p}(\omega_o|\omega_i)=\frac{\exp({v'_{\omega_o}}^{T}v_{\omega_i})}{\sum_{\omega=1}^{W}\exp({v'_{\omega}}^{T}v_{\omega_i})} \tag{3-1}$$

观察等式（3-1）的分母可以发现对于每个窗口，我们都要进行词汇表大小|W|次数的向量乘积计算，词汇表的大小通常在 10^5-10^7 范围内，需要消耗大量的时间进行网络的训练。因此，Skip-Gram 采用了层次 softmax（Hierarchical Softmax）来进行分类，层次 softmax 把一个类别数目极大的分类问题转化成了一个树结构的多层次分类问题，从而将计算的复杂度降低到了 $\log_2(|\mathrm{W}|)$，层次 softmax 的分类概率如式（3-2）所示。

$$\mathrm{p}(\omega|\omega_i)=\prod_{j=1}^{L(\omega)-1}\sigma\left([\![n(\omega,j+1)=\mathrm{ch}(n(\omega,j))]\!]\cdot {v'_{n(\omega,j)}}^{\mathrm{T}}v_{\omega_I}\right) \tag{3-2}$$

在确定了网络结构之后，只需指定使用的模型结构，中间表示向量维度，训练窗口大小即可对网络进行训练。

3.1.2 词向量生成

词向量的训练过程中只需要窗口共现信息而不需要其他的人工标注，因而我们可以大量的爬取互联网中的文本语料进行词向量的训练。考虑到本课题所构造词向量的实际应用场景是问句匹配，我们使用百度知道作为文本来源进行信息爬取，总共获取到了 20GB 百度知道问答数据。对问答数据进行初步筛选清洗之后，我们分别训练了基于词和字的词向量。

在文本分词上，目前国内很多的开源分词工具包都在中文分词领域的测评中取得了不错的效果，包括哈工大的语言技术云平台[40]、中科院 NLPIR 汉语分词系统、python 的结巴分词等。考虑到版权等因素，本课题最后采用结巴分词来对所有的语料进行分词和词性标注。

在词向量的模型及参数的选择上，使用了基本的卷积神经网络在构造的中文数据集上进行评估。在词向量的维度选择上，由目前已有的研究成果可以得知维度更高的词向量所能覆盖的语义内容更广，因而在各个任务上都能取得较好的效果，但是高维的词向量也会导致模型的参数变得复杂，在参数更新时会消耗大量的时间。出于时间因素的考虑，选择了 Skip-gram 作为训练的基本模型结构，窗口大小设置为 5，使用负采样（Negative Sampling）技术进行分类，并且选用 200 维作为词向量的维度大小。

3.2 基于卷积神经网络的语义匹配算法

与传统的机器学习方法相对比，深度学习能够自动从原始数据中获取数据特征，并且通过多层的非线性映射对多方面的特征进行组合，最后得到高层的抽象特征。这种方式大大减少了在构建传统机器学习模型时所需要的特征提取构建工作，使得研究的算法可以快速应用到实际工程上，也大大提升了模型的可移植性。因此，本课题考虑采用深度学习的方法来对问句进行语义匹配。参考目前国内外所作的工作，可以发现卷积神经网络在文本特征提取方面具有非常大的优势，使用卷积神经网络进行的情感分析、文本分类、语义匹配等工作都在公开数据集和测评上取得了非常不错的成绩，最后本课题决定使用卷积神经网络作为本课题主要研究的语义匹配算法。

3.2.1 卷积神经网络原理简述

卷积神经网络是目前深度学习领域最最要的架构之一，主要思想是由动物的视觉皮层神经感知的特点所得来的。20 世纪 90 年代，LeCun 等人进行了一系列工作，正式确立了卷积神经网络的基本结构，并且提出了一个真正在工业上应用的

模型 LeNet-5。卷积神经网络的输入通常是一个多维数组，例如一幅彩色的图像，由三个色道的二维点阵组成，很多数据的形式本身就是一种高维数组：信号和序列以及语言都是一维数组；图像和音频谱图是二维数组；视频和容积图像是三维数组。卷积神经网络的背后有四大核心思想，局部连接，权值共享，池化和多层次。一种典型的用作句子建模的卷积神经网络结构如图 3-2 所示。

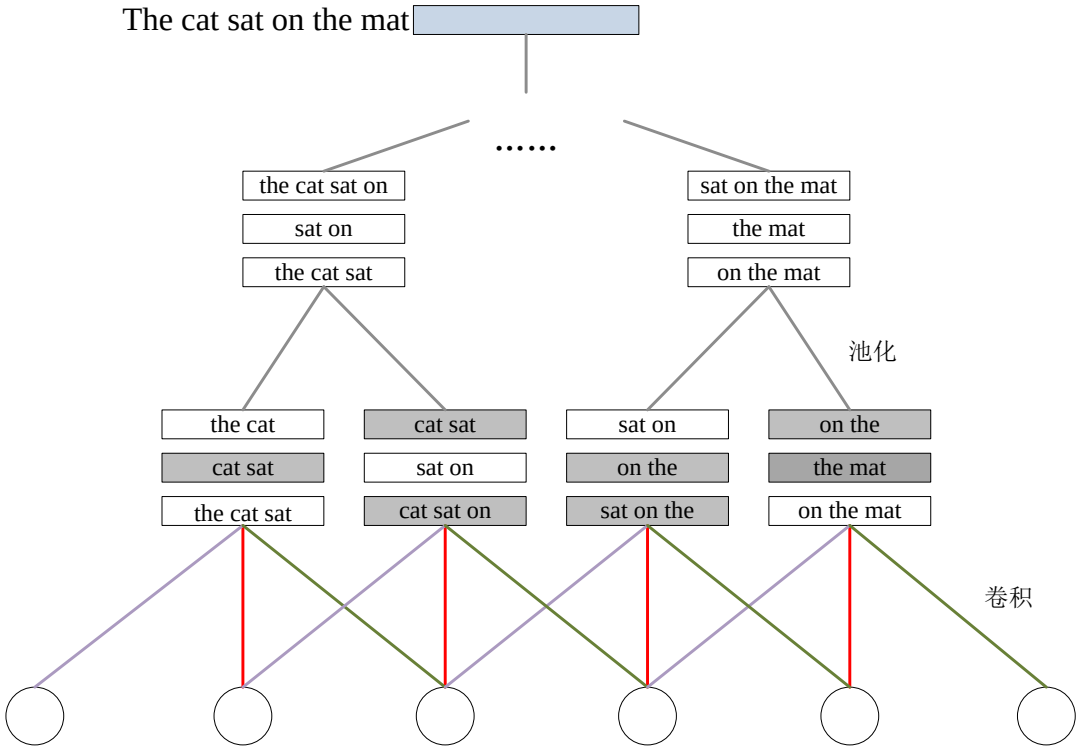

图 3-2 卷积神经网络进行句子建模结构图

卷积神经网络接收输入过后的阶段通常由卷积层和池化层交替连接组成。对卷积层的每个卷积核来说，卷积核对上一层的所有特征矩阵做卷积操作，卷积的结果将会输入到一个非线性激活单元中进行激活。卷积核通过对上一层特征矩阵的不同位置做卷积，得到了新一层的特征矩阵。卷积操作的基本公式如（3-3），下一层的特征矩阵由当前层的特征矩阵做卷积后进行一个函数映射得到。

$$x_j^l = f\left(\sum_{i \in M_j} x_i^{l-1} * k_{ij}^l + b_j^l\right) \tag{3-3}$$

在对上一层的特征矩阵进行卷积操作时，所有的滤波器组共享相同的权值，而在卷积层生成新的特征矩阵时，我们使用不同的滤波器组来进行卷积操作，这样做的原因有两个。首先，对于数据数据例如图像来说，局部群内的值通常是高

度相关的，这也就形成了区别性的局部特征主题。其次，对于图像和其他一些信号表示来说，局部的统计信息对于矩阵的所有区域通常都是不变得。换言之，如果图像的某一特定位置出现了某种图形，那么这种图形有可能在图像的任意位置出现。因此我们对特征矩阵的不同局部使用权值共享的方法进行卷积运算来检测图像中不同位置所出现的相同模式。

虽然卷积层提取到的是上一层中的局部特征信息，池化层的职责却是合并具有类似语义信息的特征。因为卷积提取到的一些形成某种模式的特征的相对位置可能会发生一些微弱的变化，因而我们通过粗糙化每个特征的位置信息来保持一种可靠的模式检测。一种典型的池化方法是计算一个特征矩阵中每一个局部块的最大值，并用这个值生成一个新的特征矩阵。相邻的池化单元通常会平移超过 1 行或一列的距离，因此这也就降低了特征表示的维度同时也创建了微弱的平移和扭曲的不变性。卷积神经网络中同样使用标准的方向传播方式对整个网络参数进行训练，并且复杂度也与其他规则的深度网络相类似，对于池化层可以计算残差如公式（3-4）。

$$\delta_j^l = \beta_j^{l+1}\left(f'\left(\mu_j^l\right)\circ \mathrm{up}\left(\delta_j^{l+1}\right)\right) \tag{3-4}$$

同样对于卷积层的梯度，使用反向传播算法进行推导，卷积之后的结果反向传播到之前的特征矩阵，可以计算该层残差如公式（3-5）所示。

$$\delta_j^l = f'\left(\mu_j^l\right)\circ \mathrm{conv2}\left(\delta_j^{l+1}, \mathrm{rot180}\left(\mathrm{k}_j^{l+1}\right), 'full'\right) \tag{3-5}$$

卷积神经网络发掘了很多自然信号中的组合层次结构（Compositional Hierarchies）信息，网络中高级的特征由低级的特征组合而成。在图像领域，局部的变的组合构成图形，图形集合成了一个部分，部分组成一个物体对象，这种类似的结构也存在于语音和文本信息中。池化操作则允许了这种上一层的特征表示有微弱位置不变性。

3.2.2 卷积神经网络输入输出处理

正如上一小节所述，卷积神经网络通常接收一个固定大小的多维数组作为输入。在本章开始部分已经讨论了使用词向量技术来讲字词转化为一个低维向量，但如何将句子转化为一个多维矩阵也是一个难点。对 2014 年以来使用卷积神经网络进行自然语言处理任务的研究成果进行调研，发现大多数研究人员直接将构成句子的词语依次放在固定矩阵的每一行上，然后对空行进行补 0（Zero Padding）操作，二维矩阵结构如图 3-3。

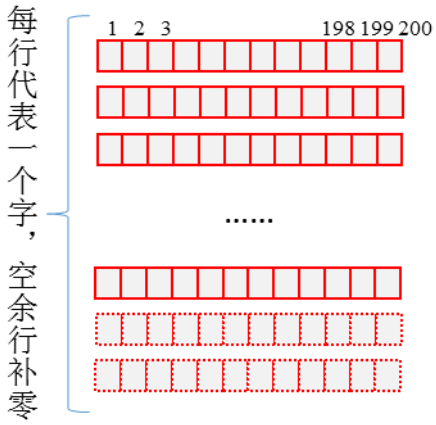

图 3-3 卷积神经网络句子输入表示

基于上述思想，可以将两个句子分别转化为二维矩阵。但是需要考虑补 0 是否将为模型引入噪声。对于卷积神经网络来说，我们认为空位补 0 是具有意义的，不会对模型本身的训练引入太大的噪声。因为卷积和池化都是一种区域性的操作，在特征为 0 的位置在卷积的过程中被新的组合特征值填充，而池化层的最大池化（Max-pooling）操作也会取当前池化窗口最大值作为特征代表，因此 0 的特征位会逐渐消失。

卷积神经网络只是对文本进行特征抽取，但是语义匹配相关性才是本课题关注的重点，因此，本课题把语义匹配问题当做一个复述识别的问题来处理，使用复述的置信度进行相关性的度量，基于这种场景，在最后一层使用逻辑回归算法进行二分类，使用类别为 1 的置信度作为语义相关性的值。逻辑回归模型是一种传统的线性二分类模型，具有训练速度快，精度较高，可导的特点。逻辑回归分类的假设函数如式（3-6）所示。

$$h_\theta(x)=\frac{1}{1+\exp\left(-\theta^T x\right)} \tag{3-6}$$

逻辑回归本身是一种优化学习算法，使用随机梯度下降对参数进行训练即可得到最后的分类器。

3.2.3 卷积神经网络结构设计

对于传统的分类任务来说，研究人员通常直接将原始数据输入到卷积神经网络中进行特征提取，再利用提取出来的组合特征应用于具体任务。而在本课题的研究任务中，卷积神经网络需要对两个句子进行处理，最后再将两个句子组合起来进行分类，因此本课题使用一种并行的卷积神经网络架构来对两个句子进行处理，网络整体结构如图 3-4 所示。

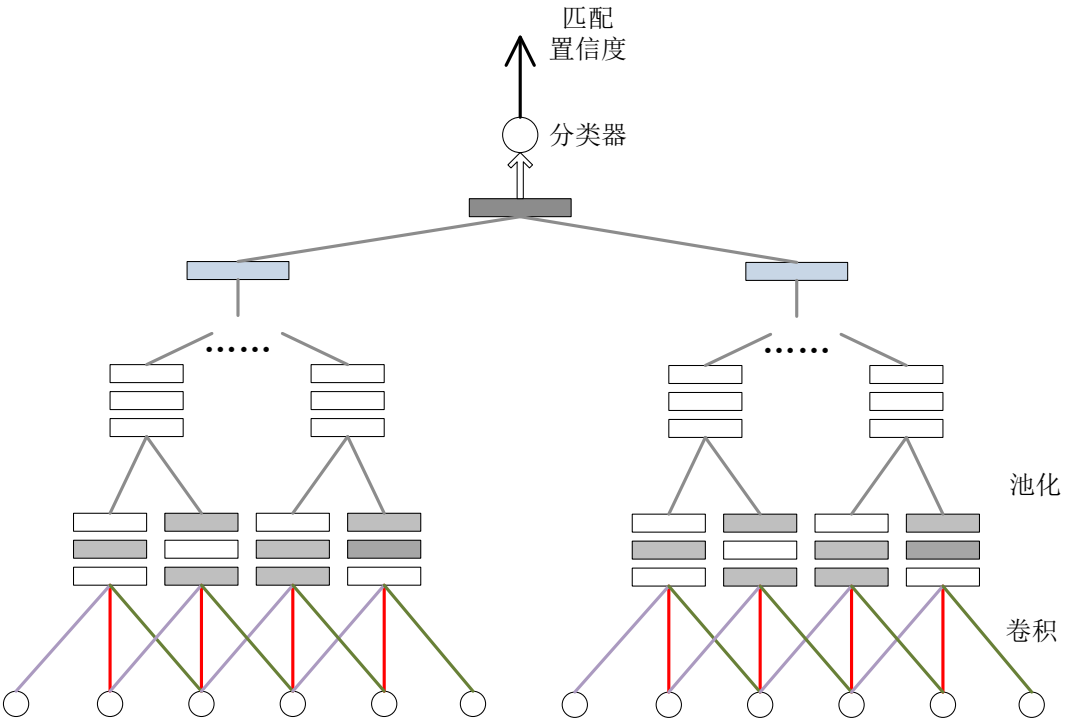

图 3-4 并行卷积神经网络结构图

可以看到与其他的卷积神经网络略有不同，本课题使用了两个并行的卷积神经网络框架分别对两个句子进行处理，每个句子的词向量组成的二维矩阵分别作为两个卷积神经网络的输入，然后卷积神经网络通过多层的卷积和池化操作提取到了高层的抽象组合特征，在最后一次卷积和池化之后对两个句子的特征向量进行拼接，再输入到一个非线性隐藏层进行非线性映射，最后输入到一个逻辑回归分类器中进行分类，网络的输出是一个 0 或 1 的类标以及分类的置信度。网络实现的具体细节由以下四部分构成：

（1）网络节点组成

卷积的层数以及卷积核的数目都会对最终结果产生一定的影响，一般来说，参数规模较大，层数较深的网络的特征提取能力较强，通常能够在测试集中取得比较不错的效果，但是复杂的模型同时也会带来训练时间的增加，过大的参数规模有可能会导致参数训练不充分、过拟合，因此考虑到模型性能与训练时间的平衡，目前使用的网络结构为：

$$\left[(1,30,200)-50C3\times 9-MP2\times 4-100C3\times 5-MP2\times 4-150C3\times 3-MP3\times 3\right]\times 2-(1,900)$$

其中$(1,30,200)$表示输入的多维数组，30 为字词的维度，200 为词向量的维度，对于句子长度大于 30 的文本采用去尾处理，对于句子长度小于 30 的文本进行补 0。

C 代表卷积层，C 前边的数字表示该层卷积核数目，C 后面接的数字表示卷积窗口大小，例如50C3×9表示卷积层由 50 个卷积核组成，每个卷积核的卷积窗口大小为3×9。MP 代表最大池化层，MP 后接的参数表示最大池化窗口大小，例如MP2×4表示池化层采用最大池化，窗口大小为2×4。网络的各个层之间使用–进行连接，中括号后的×2表示两个卷积神经网络的并行运算并对它们最后池化的结果进行拼接，得到一个1×900的特征向量。使用这个1×900的特征向量进行分类，最后得到分类结果，对于网络的层数及网络节点数目，本课题采取控制变量法设计了一系列实验进行对比，对比实验的结果及分析在第 5 章中进行介绍。

（2）激活函数选取

卷积神经网络在 1998 年就已经出现，但直到 2006 年后才真正成为机器学习领域的研究重点，这其中有很多原因，包括深度学习的计算量过大、硬件性能不够；神经网络本身比较难进行训练，经常会出现过拟合问题等。在传统的神经网络中，研究人员通常使用 sigmoid 函数作为卷积操作之后的激活函数，而 sigmoid 函数本身有一个特点就是当自变量过大或者过小的情况下，函数值变化会比较平缓从而导致导数趋近于 0，这也就会导致在深层卷积神经网络的训练过程中，经常会出现梯度消失导致参数无法及时更新，tanh 函数作为另一种常用的激活函数，同样也存在这种问题。

为了解决这一问题，我们将卷积层的激活函数改变为修正线性单元（ReLU）函数，ReLU 函数本身是一个比较简单的函数，在自变量 x 小于 0 时函数值为 0，在自变量 x 大于 0 时函数值为 x，ReLU 和 tanh 的函数曲线图如图 3-5。在其中我们可以看到当 x 大于 0 时，梯度一直保持稳定从而使我们的模型训练更为顺利。

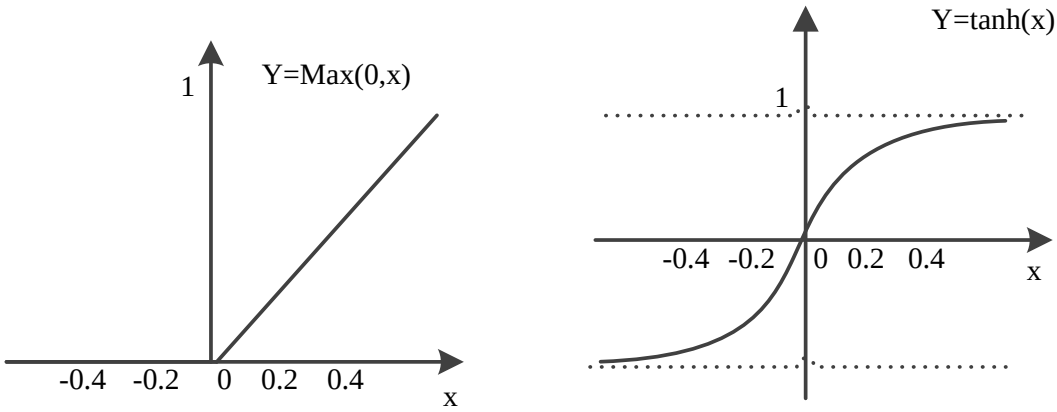

图 3-5 ReLU 与 tanh 函数曲线图对比

（3）防止过拟合

众所周知，在深度学习中有一个很大的问题就是容易过拟合的问题，特别是对于深层的卷积神经网络模型而言。由于每一层都有很多不同的卷积核导致了模型复杂度非常高也就更难于训练。在本课题中，我们引入了 dropout[41]的概念来对我们的网络进行修正。Dropout 是一种常用于卷积神经网络中的防止过拟合的方法，指的是在每个 batch 的训练过程中，dropout 机制会以一定的概率随机使得模型中的某些神经元失效，Hinton 的团队认为 dropout 其实是在每个批次的训练过程中训练了一个不同的模型，但训练出的所有模型共享同一套参数，通过这种不同模型的变换来防止了过拟合。

（4）学习率选取

训练过程中学习率如何设计也是研究人员需要考虑的一个问题。过大的学习率会导致最后得到的 cost 值过大，甚至会在反向传播的过程中出现梯度爆炸的问题，而过小的学习率则会增加网络的整体训练时间。为了平衡这一关系，本课题采用自适应学习率算法（AdaGrad）来对网络的学习率进行动态的调整，随着训练的过程不断变化，从而加快收敛。

3.3 基于注意力机制的卷积神经网络语义匹配算法

对基本的卷积神经网络语义匹配算法进行分析可以发现，前三次的卷积和池化操作对于两个句子来说都是相互独立的，模型中在分别获得了两个句子的高维特征之后对两个向量进行拼接，再输入到非线性隐藏层中进行映射，最后进行分类。由于本课题研究的是语义匹配任务，我们不能确定最后的低维向量是否能够表示句子的全部语义信息，多次的卷积和池化操作是否丢失了底层的语义信息，因此本课题在卷积神经网络中加入注意力机制来保留两个句子间的语义信息。

3.3.1 注意力机制原理简述

神经网络中的注意力（Attention）机制原理主要来源于人类视觉研究中的注意力聚焦这一过程，人类在观察事物时通常会以高注意力聚焦于某些焦点上，然后以低注意力感知周边信息，并且注意力会根据大脑的需要自动进行调整。Hinton 的团队最早在图像识别领域引入了注意力机制使用图像焦点序列进行计算，取得了与传统的使用全图信息进行计算的方法相当的效果[42]。2014 年，研究人员将注意力机制应用到自然语言处理中常用的循环神经网络中进行机器翻译的任务，取得了惊人的效果[43]。接着 Metamind 和 Facebook 的研究人员对其中的模型进行改进发展，提出了记忆网络（Memory Network）的概念用于事实类自动问答领域并

取得了不错的效果[44]。

在目前比较常见的神经机器翻译（Neural Mechine Translation，NMT）模型中，研究人员将两种语言之间的翻译视为一个编码（encoder）、解码（decoder）的过程，首先在 encoder 网络中对源语言序列进行编码，然后使用编码后的状态输入到 decoder 网络中进行解码得到目标语言，这两个过程都使用 RNN 来实现。在这种通用的 NMT 模型中，encoder 网络把源语言编码成了一个低维的特征表示，并假设这个特征表示包含全部的高层语义信息。但这种假设显然是不成立的，对于一个中等长度的句子，显然可构成这个句子的词组合是无限多的，这也导致了句子的语义是非常复杂的，一个低维的向量难以表示句子完整的语义信息，传统的 RNN 结构如图 3-6 所示。

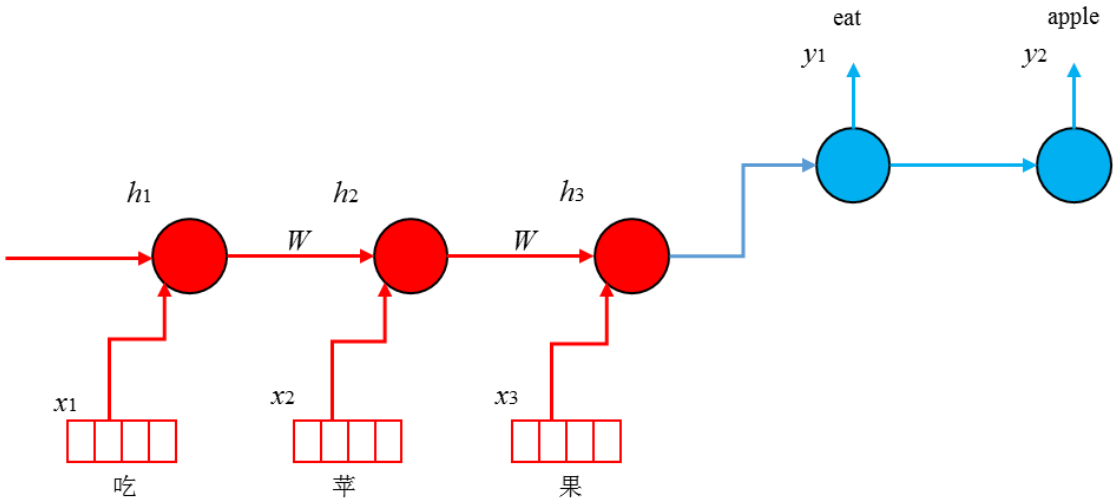

图 3-6 递归神经网络结构图

而注意力机制改变了 RNN 中的这一缺陷，我们通过使用注意力机制可以把 encoder 网络中所有的中间状态表示作为 decoder 网络的输入，并且网络会在学习的过程中自动调整参数，从而在翻译过程中动态调整每个中间状态的权值，对每个需要翻译的目的语言词语自动学习到对应的源语言词语。注意力机制原理如图 3-7，图的下半部分表示一个双向的 RNN encoder 网络，图的上半部分表示这个 RNN 结构的 decoder 网络，观察编码网络可以得知每一步所生成的隐转态在解码过程中都会加入进来，由一个注意力模块对每个隐转态的权值进行计算，根据计算得到的权值对解码过程产生影响，注意力模块的具体实现由一个浅层全连接网络构成，输入编码隐转态 h_i 和当前解码隐状态 s_t 进行运算，对各个权值进行归一化后再对每个编码隐状态进行加权。

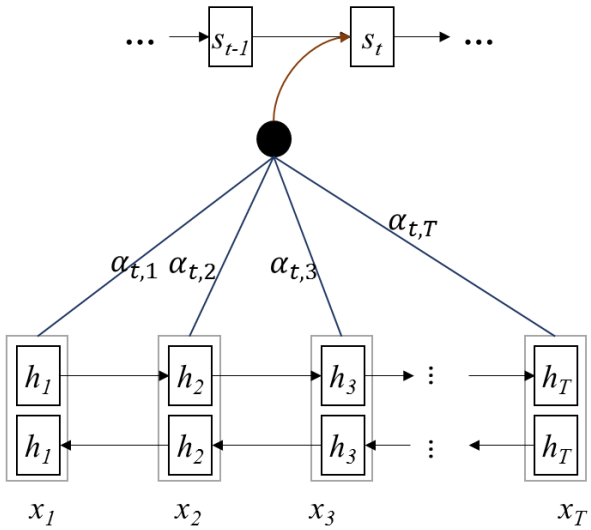

图 3-7 加入注意力机制的递归神经网络结构图

3.3.2 注意力机制模块设计

在上一小节的讨论中我们发现在基本的并行卷积神经网络中，在卷积层和池化层两个句子之间缺少了交互，而最后得到的低维向量只包含句子的高层信息，在这一过程中很可能丢失了一些句子本身的相关信息。因此基于注意力机制本身所取得的效果以及并行卷积神经网络的特点，本课题决定将注意力机制应用到并行卷积神经网络中，加强两个句子在底层特征中的交互，并学习到两个句子中的重要信息。

考虑到问句复述识别任务与机器翻译的过程本身具有一定的相似性，机器翻译的过程可以看做计算机将文本由一种语言翻译到另一种语言，而在语义匹配任务中，同样可以把两个句子的相似度视为一个句子到另一个句子的翻译概率。受到机器翻译上注意力机制的启发，本课题在每个池化层之后加入了一个改进的注意力机制来对模型结构进行修改，加入注意力机制的卷积神经网络整体结构如图 3-8 所示。

观察模型的整体结构可以发现在经过一次卷积和池化之后，模型分别对两个句子表示进行一个词语维度的最大池化操作，分别得到一个代表句子 A 和句子 B 的向量 S_A、S_B，接下来使用句子 A 的表示向量和句子 B 的所有行特征进行组合，对每个行特征得到一个语义权值 b_i，最后使用这个语义权值对行特征本身进行加权作为下一层的输入。

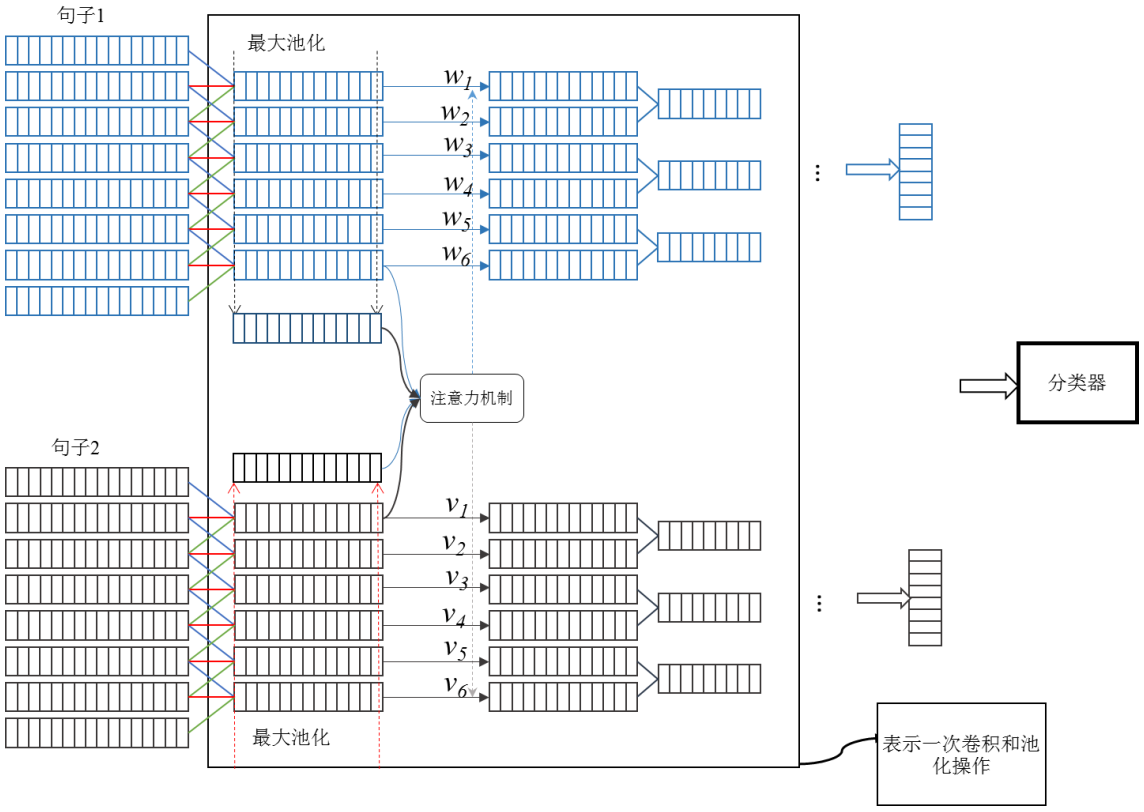

图 3-8 基于注意力机制的并行卷积神经网络结构图

对于整个注意力机制模块，首先根据式（3-7）对模块输入进行最大池化操作。

$$S_B = \mathrm{MP}\left(b_1 b_2 \ldots b_M\right) \tag{3-7}$$

接着对于得到的句子 B 表征向量 S_B，与句子 A 的行特征 a_i 进行组合进行权值学习。

$$\omega_i = f\left(\left[S_B, a_i\right]\right) \tag{3-8}$$

其中的 f 为一个多层感知机（Multilayer Perceptron，MLP），通过对 S_A 和 S_i 进行多层的特征提取及特征组合来获取行特征 S_i 的加权信息 b_i。在最后对每一个行特征进行加权之前先采用一个归一化函数来对一个句子的所有行特征权值进行归一化操作如式（3-9）所示。

$$\omega_i = \frac{\exp\left(\omega_i\right)}{\sum_{k=1}^{S}\exp\left(\omega_k\right)} \tag{3-9}$$

通过注意力操作，我们把两个句子间相关联的信息进行焦点处理，使得每一次卷积后的有用信息保留下来进入到下一层卷积层中进行运算，网络的其他模块结构与前一小节类似，注意力机制模块基本结构如图 3-9 所示。

通过在每次卷积操作后加入注意力机制进行聚焦处理，网络在底层计算时就

已经发掘了两个句子间的语义联系，通过不断的加权突出了两个句子间的相关和区别信息后得到一个有效的句子表示，因此在实际的实验中也取得了比较好的效果。

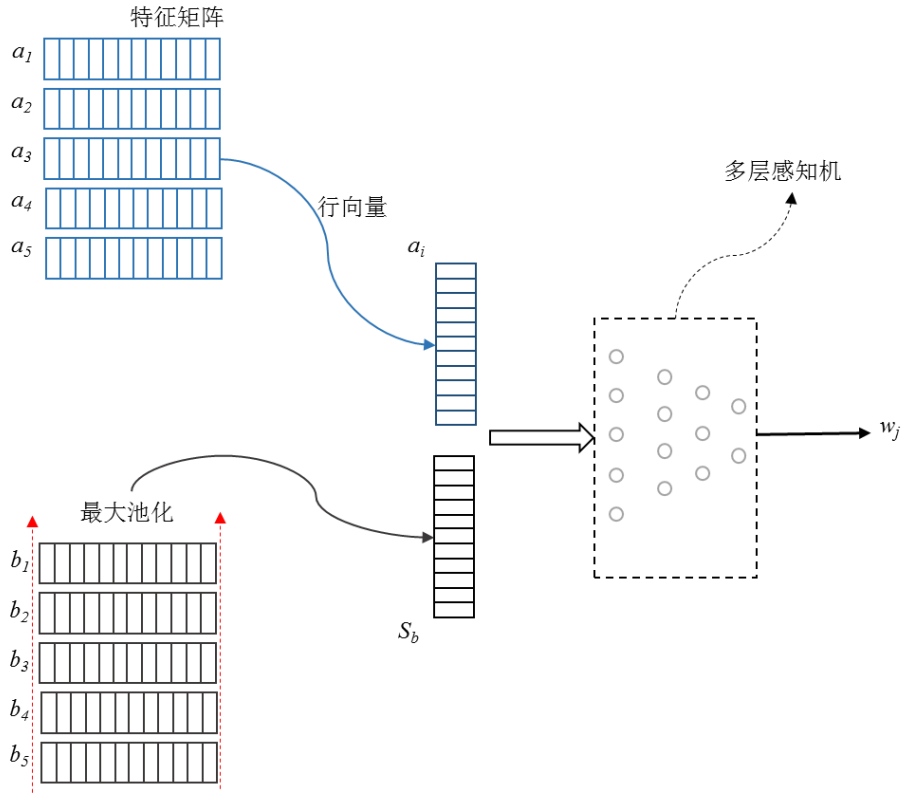

图 3-9 注意力机制模块结构图

3.4 本章小结

本章主要介绍了研究中所用到的语义匹配算法。首先介绍了在自然语言表示如何转化为神经网络输入的方法，包括传统的词袋模型以及词向量表示。接着，本章讨论了一个基本的并行卷积神经网络算法用于本课题中语义匹配的任务中，并对网络结构及网络的训练方法进行了分析。最后，本章讨论了注意力机制的原理及优点，并讲述了在本课题中如何间注意力机制与并行卷积神经网络相结合，通过注意力机制发掘句子间的相关因素和无关因素。对于实验的结果本章没有进行描述，将在第 5 章中进行分析。

第4章 自动问答系统构建

本文的前两章讨论了问答系统中语料库的构建以及核心算法模块的设计，但是语义匹配算法仅仅是问答系统中一个组成模块。对于用户以自然语言表述输入的一个句子，我们首先要对问句进行理解获取用户意图，接着根据用户意图到语义匹配模块中获取到相关的信息，最后进行答案的抽取和生成反馈给用户。同时，根据问答系统应用的领域、面向的用户、提供的作用不同，问答系统本身的组成模块也不尽相同。本章将对本课题所设计的整个问答系统构建方法做一个整体的介绍，并针对一些具体模块作出一些详细的介绍。

4.1 系统架构设计

根据之前的讨论，常用的问答系统主要由问句分析、信息检索、答案抽取和答案生成四个部分构成，但是在实际的生产环境中，各个模块的取舍通常会与实际的应用场景相关。例如，微软亚洲工程院推出的聊天机器人小冰主要是与人进行日常生活中简单的聊天，因此小冰本身的知识库主要是由互联网中提取得到的大规模语料构成，这些数据大多是开放式多领域的，而在信息检索方面，由于小冰本身是一个互联网环境下的聊天机器人，因此检索可获得的信息大多也都是实时更新的。本课题所研究的目标是构建一个针对垂直领域用户的常用问题自动问答系统，因此本课题的系统结构本身也会针对应用的垂直领域以及常用问题作出一些改进。

在实际的系统构建过程中，本课题将系统的构建过程分为系统架构和线上部署两部分，系统的整体架构图如图 4-1 所示。

问答引擎是整个问答系统的一个控制模块，主要负责系统的输入输入处理以及一些外部信息的应用。语义匹配模块包括 WMD 和 SCNN 两种算法，SCNN 代表的是文章前一部分中所构建的并行卷积神经网络模型，本课题采用离线的方式先对语义匹配模型进行训练，对模型代码进行抽象化的设计，直接提供一个匹配接口给问答引擎模块。WMD 算法代表的是词语移动距离算法，在某些训练语料不够充分的领域，SCNN 算法可能不能提供非常好的泛化效果，因此我们引入一个无监督的 WMD 算法对语义匹配的结果进行纠正。WMD 算法是一个基于词向量的概率分布相似度计算方法，因此在某些标注语料比较少的领域，我们可以使用大量的未经标注的领域文本语料来进行词向量的训练，得到一个效果比较好的词向量，再利用词向量使用概率分布距离衡量方法来对两个句子进行相似度计算。

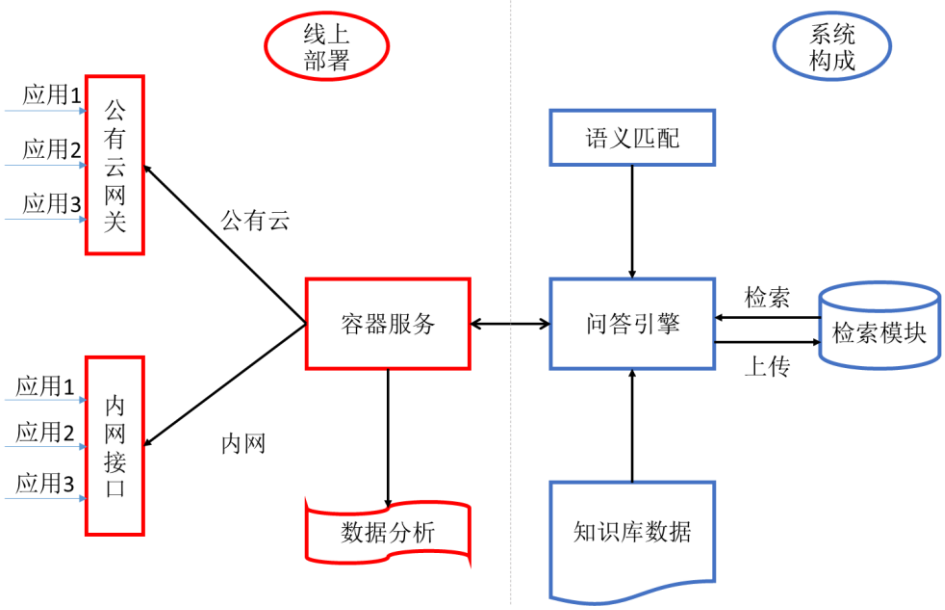

图 4-1 自动问答系统整体架构图

4.2 信息检索系统搭建

在传统的信息检索领域，用户通常会提供少量的关键词给搜索引擎，搜索引擎对这些关键词进行相似度计算从而检索得到相关文档序列。在自动问答系统中，用户输入的问句首先会输入到语义分析模块进行语义分析相关的任务，例如问答分类、主题发现、关键词提取等等，通过这些过程可以获取到代表问句语义的候选查询词以及分类、主题信息，检索系统将这些信息组合，采取不同的策略进行检索召回。例如，通常对于不同类别的信息检索系统会检索不同的知识库来获得答案。对本课题所构建的针对常用问答对的问答系统来说，检索系统需要完成的任务是找到知识库中与用户问句具有相似语义的问题列表进行进一步的语义匹配，有了问题列表之后可以很容易的索引到相关问题的答案。因此对于检索系统来说，如何度量两个问句的相似度是需要考虑的一个问题，本课题采用了两种相似度度量方式来进行相似度计算的任务。

4.2.1 向量空间模型

在传统的检索系统中，研究人员常常采用向量空间模型来表示文档。向量空间模型是一种只考虑句子内词语组成、词语浅层性质的信息检索模型，对于句子内部的语义信息和词之间的语法组合都不能很好的进行表示。但是向量空间模型由于模型构成简单，在大规模检索中的适用性非常强，并且在实际应用中也被证明有不错的效果，因此本课题使用目前效果较好的文档频率-逆文档率（Term

Frequency-Invert Document Frequency，tf-idf）来进行检索相似度计算。

tf-idf 是一种基于文档数据统计的词表示方法，如果一个词语在某个句子中的出现频率非常高，而在所有文档集合中出现次数比较少，则赋予其较高的权重。这种方法是一种词频与逆文档率相均衡的度量方法，即出现次数较多，区分度较高的词语应该在向量空间模型中具有较高的权重。首先对句子进行分词及去停用词操作，然后对句子中的每个词，计算 *TF* 值如式（4-1）所示。

$$TF_i = \frac{n_i}{\sum_k n_k} \tag{4-1}$$

公式中的 TF_i 表示第 i 个词语在句子中的词语频率，n_i 表示该词出现的次数，n_k 表示句子中第 k 个词出现的次数。

IDF 值代表的是词语对于不同文档的区分度，对于给定的一个文档集合，每个词的 *IDF* 值是固定的，具体计算方法如式（4-2）所示。

$$IDF_i = \log \frac{|D|}{|\{d, w_i \in d, d \in D\}|} \tag{4-2}$$

其中 IDF_i 表示第 i 个词的 *IDF* 值，$|D|$表示总文档个数，分母则表示包含该词的文档个数。

4.2.2 词向量余弦相似度模型

词向量的构造与训练方法在前一章中已经进行了阐述，词向量是对自然语言中词语的一种低维度、稠密的向量表示方法，通过对训练生成的词向量进行分析，研究人员发现词向量本身存在着一种语言类比的特性。例如，“中国”的词向量与“首都”的词向量执行相加操作后与“北京”的词向量非常接近，“女人”的词向量与“国王”的词向量执行相加操作后与“女王”的词向量非常接近，这也就意味着对于词向量执行一些基本的数学运算操作是合理的，因而在本课题中直接将短文本句子表示为句子中词向量算术和的形式。

对于两个由向量表示的文档来说，可以使用余弦相似度来度量两个句子间的距离，如式（3-12）所示。

$$sim(\boldsymbol{w}_1, \boldsymbol{w}_2) = \frac{\boldsymbol{w}_1 \cdot \boldsymbol{w}_2}{\|\boldsymbol{w}_1\| \times \|\boldsymbol{w}_2\|} = \frac{\sum_i p_i \times q_i}{\sqrt{\sum_i p_i^2 \times \sum_i q_i^2}} \tag{4-3}$$

其中 w_1、w_2 分别表示两个句子，p_i 和 q_i 分别表示两个句子向量中的第 i 个分量。

4.2.3 检索系统搭建

在实际检索系统的构建过程中，目前互联网上已经有比较多开源的检索引擎

可供选择，例如 Lucene、Sphinx、Xapian 等等，这些检索引擎通常都已经构造的比较成熟，开发人员数据进行简单处理然后调用已经写好的接口即可使用检索服务，本课题中出于数据的安全性以及检索速度的考虑，我们使用阿里云提供的开放搜索（OpenSearch）服务来对检索模块进行构建，OpenSearch 是由阿里巴巴搜索团队自主开发的一个大规模分布式搜索引擎平台，开发人员可以在阿里云上购买该服务并且很方便的进行使用。

4.3 问答系统容器化

4.3.1 容器服务简述

容器服务（docker）是近几年兴起的一种轻量级虚拟化技术，通过软件级别的操作系统虚拟化使得我们的一台硬件机以器上可运行多个运行环境相冲突的应用，docker 本身具有虚拟化、轻量级、秒级启动、开发运维分离等特点，依托于近几年出现的云计算服务中的基本概念研究人员推出了容器即服务（Docker as a Service，DAAS）的概念，使得我们可以在目前的主流云平台上方便的使用容器技术对系统进行部署，使用容器服务进行部署使得系统的开发部署周期大大缩短，测试人员可以直接使用开发人员构建的镜像镜像测试，系统的维护也变得更为方便，使用容器服务进行部署的方式如图 4-2 所示。

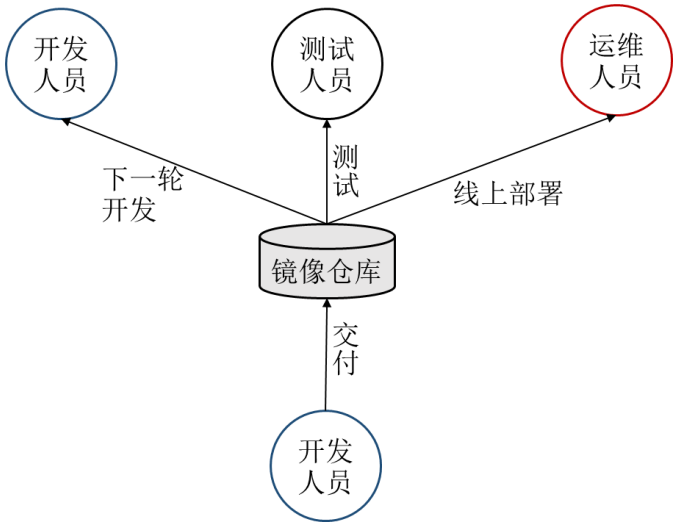

图 4-2 使用容器技术进行线上部署结构图

4.3.2 问答服务容器化

由于本课题所构建的问答系统的依赖环境比较复杂，而不同领域的客户也对本系统的可移植性产生了比较高的要求，因此本课题使用容器服务来进行系统的线上部署工作，通过对问答系统的运行环境单独构建的镜像实现了环境与代码的

解耦，我们只需要对整个问答系统依赖的环境进行一次离线的构建，然后将环境镜像上传到私有镜像仓库中即可进行重复使用。由于系统本身依赖的安装环境源大多在国外，原本进行一次系统的线上部署通常需要几个小时的时间才能部署成功，在网络条件不佳的情况下甚至可能部署失败，在使用容器技术之后只需十分钟时间系统即可成功部署，使用容器技术部署问答系统的过程如图 4-3 所示。

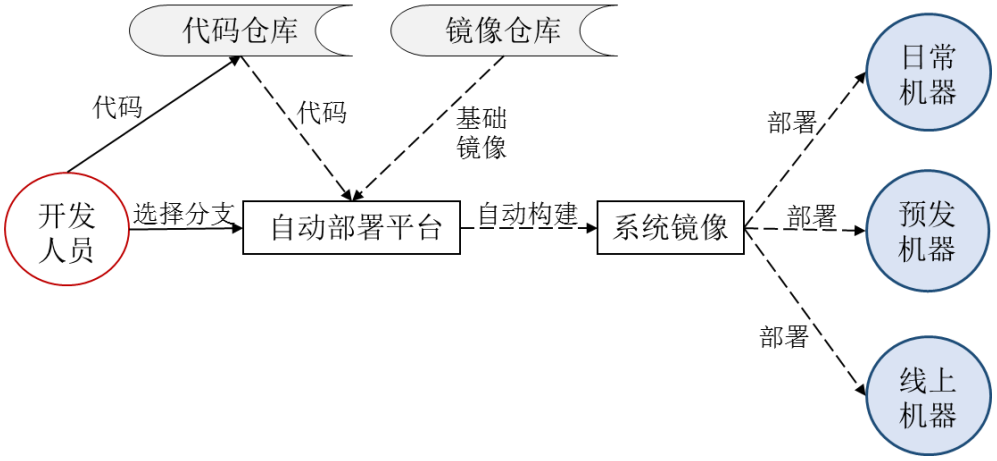

图 4-3 使用容器技术部署问答系统的整体流程图

4.3.3 数据分析服务

对于一个线上实际运行的系统来说，每天所产生的数据量是非常大的，通常开发人员使用日志来记录系统的访问情况以及重要的中间信息。对于系统的运维人员来说，系统日志可以帮助他们快速的定位系统出现的问题，及时修复系统漏洞；对于算法开发人员来说，通过对系统的输入、输出以及用户反馈的统计，开发人员可以比较容易的分析出系统中算法的缺陷并进行合理改进。开发人员通过对用户反馈信息进行统计，可以很容易的提取出系统返回的不好的案例从而进行分析改进，在数据分析阶段，本课题采用阿里云计算提供的大数据计算（MaxCompute）服务来进行日志数据的统计分析。首先，定期的对系统运行日志进行上传工作，并在 MaxCompute 上对日志进行结构化处理存储为表的形式，此时开发人员就可以在阿里云的页面上观测到可视化的统计信息；对于一些特定的数据分析需求，开发人员只需编写 SQL 脚本进行数据的统计；更进一步，MaxCompute 实现了大量的机器学习、数据挖掘方法，开发人员可以直接调用 API 接口对算法进行训练。

4.4 本章小结

本章详细介绍了问答系统的整体实现，包括系统构成和线上部署两个方面。

首先对系统的整体架构进行了介绍，包括系统的问答引擎、信息检索、语义匹配算法几个基本模块以及问答系统容器化，数据分析服务等线上部署模块。用户提供的问句首先由问答引擎模块进行处理，接下来根据分析得到的信息输入到检索模块进行检索并返回候选结果，对于返回的候选结果列表使用语义匹配算法模块进行进一步的筛选最后返回正确结果给用户。对于信息检索模块，本章主要介绍了两种语义相似度的衡量方法以及检索引擎的具体实现。此外，在问答系统的上线过程中，本章详细介绍了如何使用容器技术来进行问答系统的部署以及如何使用数据分析服务对系统日志进行分析。

第5章 实验结果分析

本文的前 4 章主要介绍了问答系统中语义匹配算法的设计以及系统的整体架构，本章节将对前面所涉及到的实验结果进行详细的分析及描述。本章的第一部分主要讨论语料集构造相关的实验结果，包括种子选取策略、筛选及算法验证三个方面；在本章第二部分，主要讨论不同算法模型的对比实验；本章的最后一部分主要描述问答系统在实际应用中的使用情况。

5.1 评估指标介绍

本课题的实验部分主要对各个方法的性能以及时间复杂度进行了对比统计。对于使用神经网络的各个模型，本课题采用训练时间作为指标来对比各个模型的时间复杂度。而对于实验中使用的各个算法，本课题主要采用了四个指标来对算法的性能进行衡量。

（1）准确率（Precision，Prec）

在本课题中，准确率表示算法分类得到的正确正例数与算法分类得到的总正例数之间的比例，计算公式如式（5-1）。

$$\mathrm{Prec} = \frac{N_{tp}}{N_p} \tag{5-1}$$

式中 N_{tp} ——算法进行分类得到的正确正例数；

N_p ——算法进行分裂得到的总正例数。

（2）召回率（Recall，Rec）

在本课题中，召回率表示算法分类得到的正确正例数与数据集中总正例数之间的比例，计算公式如式（5-2）。

$$\mathrm{Rec} = \frac{N_{tp}}{N_t} \tag{5-2}$$

式中 N_t ——算法进行分类得到的正确正例数。

（3）F1 值

F1 值代表着 Prec 与 Rec 之间的均衡，通常用 F1 值作为评估算法性能的综合指标，计算公式如式（5-3）。

$$\mathrm{F1} = \frac{2 \times Prec \times Rec}{Prec + Rec} \tag{5-3}$$

（4）总精度（Accuracy，Acc）

总精度表示算法划分出的正负例与实际数据集中样本类标相比的总准确率，计算公式如式（5-4）。

$$Acc = \frac{N_{tp} + N_{t}}{N} \tag{5-4}$$

式中 N_{tn}——算法进行分类得到的正确负例数；

N ——数据集中的样本总数。

5.2 语料集构造

5.2.1 语料集来源

本研究的实验数据主要来源于互联网的在线社区问答，其中主要由百度知道构成。对于百度知道的每一个条目，我们爬取了问题简述、问题详细描述以及最佳答案，其中问题简述用于第 2 章所描述的问句复述识别语料库的构造。百度知道数据主要包括最初的 90 万种子问句，由种子问句进行搜索爬取得到 9000 万个问答条目，通过第 2 章的数据筛选方法最后得到了 240660 条人工标注问句对数据以及 500000 条自动标注问句对数据。这一部分的实验主要目的是验证相关的抽取和筛选方法是否有效，以及对最后构造的语料集进行评估。

5.2.2 种子选取策略对比

在第 2 章的语料集获取小节中，首先介绍了爬虫种子选取的几种策略。第一种方案是直接使用搜狗细胞顶层词库的词汇进行问答条目爬取；第二种方案是通过一个自扩展迭代的流程来增加关键词的个数，使用多个组合关键词进行问答条目爬取；第三种方案是使用一些筛选过后的种子问句进行问答条目爬取。在这一小节中将详细分析这几种方案所获得的结果，并作出比较。

第一种方案中使用单关键字作为种子词进行爬取，这种方法获得的结果发散性较强，在同一个词的结果中通常具有非常多种类的语义表示；第二种方案中使用多关键字作为种子词进行爬取，多个关键字的约束使得检索获得的句子列表语义较为集中，召回的 100 个问句中基本上集中于几种语义表示；第三种方案中使用整句作为种子进行爬取，整句约束使得检索返回的问句列表语义大部分与种子句子非常接近，用三种方案分别进行检索爬取示例如图 5-1。图中的三个矩形框内分别列出了使用“购物”作为种子、使用“购物 网站”作为种子、使用“怎样做好一个购物网站”作为种子进行检索返回的前五个问句。对三个例子进行分析可以发现；单查询词返回的五个问句分别表示了四种意思，表述方式都比较不一致；

多查询词返回的句子受到两个关键词的约束，返回的表示比较集中，但在语义上也有比较大的区别；整句查询返回的句子语义上都与种子句比较接近，语义上有一些差别的句子例如“怎么用 html 做购物网站”比较容易在后面的筛选过程中进行排除。

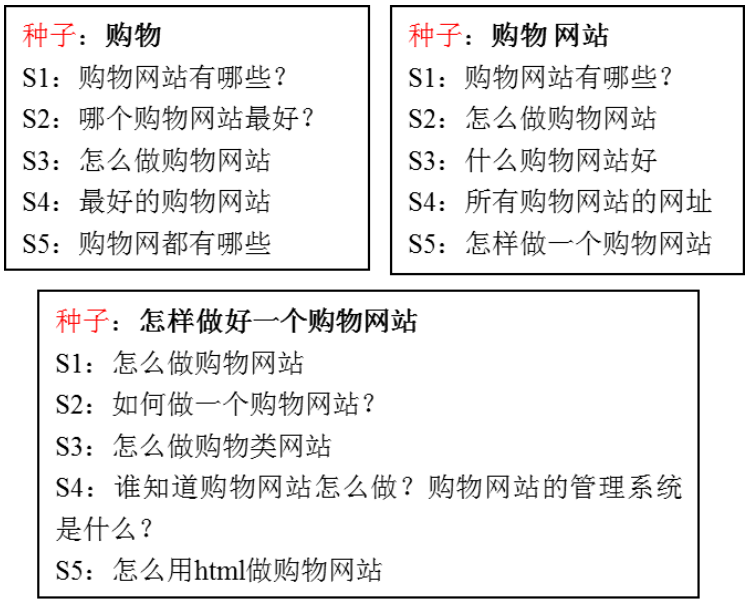

图 5-1 三种种子选取方案的检索结果示例

对于本课题的任务来说，我们希望通过一系列自动化的工作来获取精度比较高的正例样本（负例随机构造），这样在人工标注的阶段可以标注规模较小的数据减少人为的工作量，而自动构造的高精度样本同时可以帮助模型进行训练，基于这一目标，针对三种种子选取策略所返回的结果，整句的选取策略使得语义比较集中，在这一步中取得了比较好的效果。

5.2.3 问句对抽取方法对比

在获得了一个大规模的种子-问句列表集合之后，我们需要采取一定的策略来构建复述识别的问句对。在第二种中提到了三种策略来进行问句对的抽取，第一种是使用 k-means 聚类算法来对每一个种子的问句列表进行聚类操作，在每个类内部随机组合构成问句对，第二种是使用 tf-idf 进行关键词抽取，对具有相同关键词的问句随机组合构成问句对，第三种是直接使用检索召回的前 5 个问句分别与种子问句进行组合，再使用一些模式匹配的方法来对这些问句对进行筛选，在这一小节中将对这些方法进行比较。

本小节的部分数据采用人工标注进行验证，因此先简单对人工标注的流程进行介绍。对于每个问句对来说，标注人员可以将该问句对标为 1、0、N 三种形式，

其中 1 代表标注人员认为两个问句表示相同的意思，0 代表标注人员认为两个问句表示不同的意思，N 则代表标注人员无法确定两个句子间的关联性。对于每个需要标注的问句对都会由三个不同的标注人员进行标注工作，在标注完成后对所有问句的标注标签进行统计，假如某个标签出现超过两次，那么这个标签将会作为数据标注的最终标签，而每种标签出现一次的数据将被丢弃并进行分析，看是否是标注人员对问题的理解出现了问题。因此对最后人工标注的数据存在 1、0、N 三种类标。

对三种筛选方法，本课题对第一种和第三种方法进行了抽样并提交给外包人员进行标注，对第二种方法进行了抽样由开发人员进行了标注，标注结果如表 5-1 所示。

表 5-1 三种问句筛选方法对比

筛选方法	标注正例	标注负例	Prec(%)
K-means 聚类	9725	19348	33.4
关键字聚类	633	367	63.3
模式匹配	129725	47839	**73.1**

表中的 Prec 表示标注正例数量占总样本数量的比例，也就是精度（Precision），在最终统计过程中对标注成 N 的问句对进行丢弃，可以发现使用模式匹配方法来进行问句对抽取具有最好的效果。

本文使用了两种强度的模式匹配方法对问句对进行筛选。第一种模式匹配方法将主语，宾语和动词完全相同的两个句子分为正例；第二种模式匹配方法只要求两个句子的主语和宾语比较相似，并使用 word2vec 来做词的泛化。第一种模式匹配方法的匹配规则非常严格，因此可以获得高精度的正例但是两个句子之间缺少变化，而第二种方法可以发现更多的具有不同句型和词形表示的问句对，取一例子分析如表 5-2 所示。

表 5-2 两种匹配规则筛选结果举例

匹配规则 1	匹配规则 2
句子 1：自己在家怎么制作南瓜饼？	句子 1：自己在家怎么制作南瓜饼？
句子 2：在家怎么做南瓜饼！	句子 2：好想知道在家如何做南瓜饼！

在执行完所有操作之后可以得到候选问句对，这之后还需要使用 WMD 算法进行进一步的筛选，原因如图 5-2 所示。图中展示了两个由传统的过滤方法筛选之后的例子，可以发现在使用传统的过滤方法进行筛选之后，候选问句对中还是混

杂着比较多的负例，这也就导致了自动构造的训练集质量非常低，因此需要使用 WMD 算法进行最后一步的过滤。

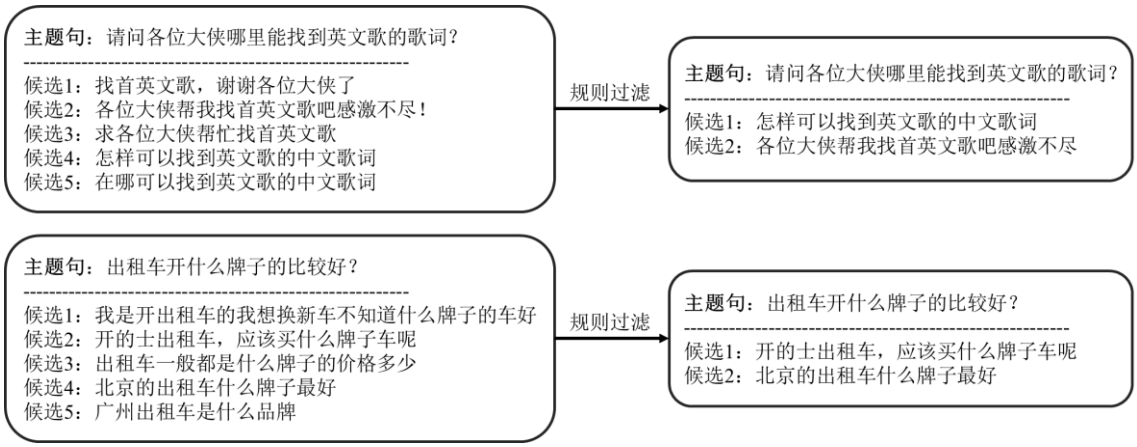

图 5-2 使用模式匹配方式进行筛选结果示例

5.2.4 语料质量评估

在进行所有的收集和筛选工作后，本课题随机抽取出一部分进问句对行人工标注工作并对剩余的问句对使用算法进行进一步筛选生成自动构造的问句对，数据的构造已经在第 2 章中做出了介绍，可以发现数据集由人工标注训练集、自动标注训练集、验证集、测试集四部分构成，其中除了自动标注训练集之外的所有样本都由人工标注获得，验证集和测试集正负样本比都是 1:1。

本课题采用了两种算法来验证人工标注数据集的质量，第一种算法是 WMD 算法，这种距离度量算法在 2015 年的文本聚类任务中取得了历史最好的成绩；第二种算法是本课题第 3 章介绍的并行卷积神经网络模型；两种算法都是用单字处理，其中第一种是无监督算法，第二种有监督算法，表现如下。其中，Prec 表示正例的准确率；Rec 表示正例的召回率；F1 值为正例的准确率和召回率的一种加权指标；Acc 表示所有样本的准确率。可以发现人工标注训练数据训练的 CNN 模型在各个指标上都高于 WMD 模型，这也就验证了构造的标注训练集对于有监督学习是有效的。

表 5-3 使用人工标注语料集进行训练结果对比

模型	Prec(%)	Rec(%)	F1(%)	Acc(%)
WMD	67.0	81.2	73.4	70.6
CNN	**70.3**	**82.9**	**76.1**	**73.9**

对于自动构造的数据，本课题选择了单独使用和加入到人工标注数据中两种

方式使用单字 CNN 模型进行训练，实验结果如下。可以发现，虽然单独训练的结果不是很好，但是加入到人工标注的训练集中却使得结果有了较大的提升，增大了正例的召回率以及 F1 值，虽然使得整体的准确度有了一定的下降，但也意味着自动构造的训练样本也是有意义的，在下文的对比实验中，未经注释的情况下均使用 74w 样本进行训练。

表 5-4 使用不同的训练数据进行训练结果对比

训练样本	Prec(%)	Rec(%)	F1(%)	Acc(%)
人工标注(24w)	**70.3**	82.9	76.1	**73.9**
自动标注(50w)	57.5	87.8	69.5	61.5
人工+自动(74w)	67.8	**89.7**	**77.3**	73.6

5.3 基础算法对比

5.3.1 实验环境

本课题的大部分实验均于阿里巴巴云计算（北京）完成，实验的主要机器是公司提供的 GPU 服务器，机器配置如表 5-5 所示。

表 5-5 实验机器配置清单

服务器	CPU	GPU	RAM	操作系统
Tesla	Intel® Xeon® processor E5-2600	NVIDIA TESLA K10 x 8	32GB	CentOS 5.7

由于深度学习模型的参数较多，计算量较大，本课题中采用 GPU 来对各个模型进行训练，并且通过多 GPU 并行计算来加快模型的训练速度，训练单字的 CNN 模型时间统计如表 5-6，后面的模型训练均使用多 GPU 方式实现。

表 5-6 不同 GPU 数量对训练时间的影响

GPU 数量	单轮时间（分钟）	训练总时间（天）	加速比（单 GPU）
1	120	7	1
4	35	2	3.4

5.3.2 卷积结构调整

与英文等一些以空格划分词的语言不同，中文通常表示为一串连续子序列，需要使用一些分词算法进行词语的划分，因此在模型的训练过程中，可以使用单

字或者分词的方式对句子进行处理，使用两种模型对两种句子表示方式进行验证，结果如表 5-7 所示。可以发现在两种模型上单字都取得了比较好的效果，原因可能是中文分词的过程可能会引入一些分词错误，而且深层的神经网络本身能发掘字与字之间的组合信息，因而分词变得不那么重要，在下文中未经注释情况下均使用单字方式表示句子。

表 5-7 使用不同句子表示形式训练模型的语料集对比结果

模型	句子表示形式	Prec(%)	Rec(%)	F1(%)	Acc(%)
WMD	单字	67.0	81.2	73.4	70.6
	分词	64.4	78.6	70.8	60.0
CNN	单字	**67.8**	**89.7**	**77.3**	**73.6**
	分词	61.7	89.4	73.0	68.3

由于 CNN 模型中的参数规模是人为设置的，本课题对比了三种卷积结构的 CNN 模型，每种结构的卷积核数量分别为 $(100,200)$ 、 $(100,150,200)$ 和 $(100,150,150,200)$，2 层和 3 层卷积结构使用标准的卷积+池化结构，而 4 层卷积则使用 2 层卷积+池化结构，结果如表 5-8。在这里之所以采用浅层卷积的原因有以下两个。首先，从目前自然语言处理中已有的实验来看，绝大部分任务中浅层卷积可以获得较好的效果，深层卷积不会对结果产生较好的影响。其次，自然语言本身具有稀疏性，而句子本身的特征矩阵表示也受到词数的限制，这一点是与图像处理中大不相同的。观察发现在使用 3 层卷积结构之后对卷积层数和卷积核数进行增加对模型的性能不会产生太大影响，但是却使得模型的训练时间大大增加。而相比 2 层结构的卷积结构，具有 3 层卷积的 CNN 模型在性能上有比较大的提升。分析其原因可能是在自然语言任务中与图像的输入不同，句子转化成的特征矩阵的行数和列数一般是不等的，矩阵的列数受限于句子的字数通常限定为 30，较浅层的神经网络足以发掘其中的关联语义信息，本课题最终使用 3 层卷积作为实验模型结构。

表 5-8 使用不同卷积结构训练模型的语料集对比结果

模型	Prec(%)	Rec(%)	F1(%)	Acc(%)
CNN（4 层卷积）	**68.1**	89.2	77.2	73.6
CNN（3 层卷积）	67.8	**89.7**	**77.3**	**73.6**
CNN（2 层卷积）	65.9	89.0	75.7	72.1

5.3.3 基础算法对比

在确定了卷积模型结构之后，本课题同时实现了一个在建立语言模型中比较常用的循环神经网络-长短时记忆（LSTM）模型来进行对比。循环神经网络由于输入是不定长的序列而被广泛用于自然语言处理任务中语言模型的建立[45]，而近年来提出的 LSTM、GRU 等结构也对循环神经网络本身的缺陷进行了改进。对比实验使用 WMD 算法作为基线（Baseline），实验结果如表 5-9 所示。可以发现 CNN 结构在正例准确率、F1 值、总样本准确率上都取得了最好的效果，LSTM 模型获得了最高的召回率但在其他的方面都比较低，原因可能是语义匹配任务需要发掘句子的深层语义信息，而多层的卷积神经网络可以比较好的做到这一点。

表 5-9 使用不同语义匹配算法的语料集对比结果

模型	Prec(%)	Rec(%)	F1(%)	Acc(%)
WMD(baseline)	67.0	81.2	73.4	70.6
CNN	**67.8**	89.7	**77.3**	**73.6**
LSTM	60.4	**90.7**	72.5	65.6

5.3.4 标准数据集验证

由于本课题的总目标是构建一个中文自动问答系统，因此在数据集构造的过程中使用中文语料作为原始文本，但是这也就导致了模型无法在英文的公开数据集上进行对比验证，因此本课题使用人工的方法将 MSRP 语料集翻译成了中文。MSRP 语料库是微软研究院在 2004 年发布的复述识别公开语料库，语料库总大小为 5799 个样本，其中包含 4076 条训练集以及 1725 条测试集。训练集中的正例数目为 2753 条，占训练集样本总数的 67.5%；测试集中的正例数目为 1147 条，占测试集样本总数的 66.5%。值得一提的是该语料集是由平行的新闻语料构建而成的，文本长度通常在 20-50 个单词之间，并且所有样本的两个句子之间都有较强的相似性，识别难度比较大。在该数据集中，研究人员一般定义将所有样本全部分为正例的模型为语料集的 baseline。

为了与其他算法进行公平的比较，本课题使用人工的方法将 MSRP 语料集翻译成了中文并使用人工分割的方法将语料集分割成了短问句对，最后使用 CNN 模型在该数据集上进行测试，结果如表 5-10 所示。其中 sub15 代表将本课题构造的句子长度大于 15 个字的问句对加入到 MSRP 的训练集中，总共为 6500 对样本，其中包括 3458 对正例，3042 对负例；sub20 代表将本课题构造的句子长度大于 20 个字的问句对加入到 MSRP 的训练集中，总共为 2837 对样本，其中包括 1544 对

正例，1393 对负例；Baseline 表示模型将所有测试集样本识别为正例。可以发现使用 sub20 数据加入到 MSRP 语料中进行取得了最好的 F1 值，这也从另一方面证明了本课题构造的数据是有效的。

表 5-10 模型在 MSRP 语料上的对比结果

模型	Prec(%)	Rec(%)	F1(%)	Acc(%)
baseline	66.5	**100.0**	79.9	**66.5**
CNN(MSRP)	**69.5**	94.5	80.1	65.4
CNN(MSRP+sub15)	68.6	97.5	80.5	65.3
CNN(MSRP+sub20)	68.6	98.3	**80.8**	65.5

5.4 模型改进实验

在确立了基本的卷积神经网络结构之后本课题使用注意力机制来对基本卷积结构进行改进，由第 4 章的分析可以得知注意力机制模块可以加入到卷积神经网络中不同的地方，并且在每一层是否都需要加入注意力机制也是我们需要考虑的问题，对于最后确定的 3 层卷积结构，本课题分别在每一层卷积之后加入注意力机制，实验结果如表 5-11 所示。表中的 1 层 Attention 代表只在第一层卷积之后加入注意力机制，2 层 Attention 代表分别在第一层和第二层卷积之后加入注意力机制，3 层 Attention 代表分别在三层卷积之后加入注意力机制。之所以从第一层就开始加入注意力机制主要是因为模型改进的目的主要是来发现句子之间底层的语义关联性，可以认为在第一次卷积的时候句子的特征矩阵主要由词之间的底层特征构成，因而注意力机制由低层向高层依次加入。可以发现注意力机制的加入确实让模型在数据集上取得了更好的效果。

表 5-11 模型使用不同层数注意力机制的语料集对比结果

模型	Prec(%)	Rec(%)	F1(%)	Acc(%)
CNN	67.8	89.7	77.3	73.6
CNN+1 层 Attention	68.5	90.4	77.9	73.7
CNN+2 层 Attention	68.6	90.5	78.0	73.9
CNN+3 层 Attention	**68.9**	**90.5**	**78.3**	**74.2**

根据上表可以得知，加入 3 层注意力机制可以让模型获得最大的提升，但是注意力机制的加入同样也意味着模型训练时间的增加，因此我们对三种模型改进

方法所需的训练时间进行统计，结果如表 5-12。可以发现第一层的注意力机制使得模型在的复杂度提升了很多，训练时间达到了普通 CNN 结构的两倍，而第二层和第三层的注意力机制并没有增加太多的训练时间，这一点主要是因为底层的特征矩阵的行数和列数都是比较大的，而注意力机制需要对每个特征矩阵逐行进行处理，这一部分的计算量是非常大的，随着卷积和池化操作的运算，特征矩阵的维度越来越小，从而注意力模块的计算量也逐步降低。

表 5-12 模型使用不同层数注意力机制的训练时间

模型	单轮时间（分钟）	总时间（小时）
CNN	35	44
CNN+1 层 Attention	62	80
CNN+2 层 Attention	70	87
CNN+3 层 Attention	72	90

5.5 系统性能

依照前面几章介绍的整体架构，本课题最终构造了一个针对电子商务领域的常见问题自动问答系统。系统的知识库由人工构建，知识库内存储着一些电商领域常用的问答对，对于用户输入的自然语言表述的问题，系统首先对问句进行分析，然后进入到检索模块进行候选答案召回，最后使用语义匹配算法获得最相关的问句并将对应的答案返回给用户。

对于实际运行系统来说，如何评估系统性能也是需要考虑的一个问题。针对实际应用场景，本课题使用 Prec 来评估系统性能的好坏，Prec 指标代表着系统返回的第一条答案为正确答案的概率。出于对比评估的考虑，本课题使用网页搜索中常用的文档相似度度量方法 BM25 算法作为 baseline，在应用场景中抽取了 1000 条用户问题进行提问，并以人工的方式对答案进行评估，结果如表 5-13 所示。

表 5-13 自动问答系统实际环境下测试对比结果

系统	Prec(%)
BM25(baseline)	63.9
自动问答系统	**84.7**

5.6 本章小结

本章主要介绍了整个实验过程中数据集构建相关实验、算法设计相关的对比

实验以及实际系统的性能。其中，数据集的构建包括种子选取方法的对比和筛选方法的对比，以及使用人工标注和模型训练的方法来评估数据集的质量。在算法相关的实验中首先介绍了实验环境以及 GPU 并行训练，然后介绍了基础算法实验和改进算法实验两大部分。前者主要讨论了卷积神经网络的基本框架对比以及在标准数据集上与其它方法的对比，包括 WMD 算法和 LSTM 算法。后者讨论了注意力机制加入对模型产生的影响，包括时间上的和性能上的，最后确定了一个效果最好的模型作为系统的语义匹配算法。本章的最后一部分讨论了整个系统在实际应用中的性能，发现本课题构造的问答系统效果远远好于传统的网页检索。

结　论

本课题主要针对垂直领域的 FAQ 自动问答系统提出了一套比较有效的构建方法，将自动问答系统分解成了问句分析、信息检索、语义匹配、知识库构建四个模块来对用户的输入进行反馈，并在各个模块中引入相应的自然语言处理技术提高准确率。对于系统中的语义匹配模块，本课题实现了多个目前比较常用的方法进行性能上的对比，继而提出了一种新型的改进方法，并使用对比实验对新型方法的有效性进行了验证。在系统的性能评估实验中也可以发现比起传统的使用信息检索进行召回，本课题所构建的问答系统确实在精度上取得了比较大的提升。

本课题的研究内容主要包括数据集构建、语义匹配算法设计、问答系统构建三个方面：

（1）本课题对目前已有的语义匹配开放数据集进行分析，构建了一个大规模的中文问句复述数据集。为了提高构建数据集的质量，在构建数据集的过程中使用了多种种子选取策略以及筛选策略进行对比，并使用人工标注和自动标注的方法分别构造了两套训练集，对整个数据集的构造流程进行了完整的介绍。

（2）本课题对目前使用较为普遍的语义匹配算法进行分析，针对中文问句匹配的任务实现了 WMD、CNN、LSTM 三种模型进行对比，并且对算法的结构也作出了具体的调整。对于卷积神经网络，本课题提出了一种改进的注意力机制加入到卷积神经网络的卷积结构中，使用实验进行验证取得了优于其它几种方法的结果。

（3）本课题对自动问答系统的整体架构进行了简单介绍，详细分析了系统中的语义匹配、信息检索以及 docker 部署模块。其中语义匹配模块主要是使用本课题构建的的卷积神经网络模型以及 WMD 模型对用户问句和候选问句进行语义匹配，信息检索模块负责根据用户输入返回候选问句列表，而整个问答系统在阿里云上使用 docker 进行线上部署工作。

除此之外，本课题针对电商领域构建的一个自动问答系统已经进入到开发测试阶段，在测试阶段取得了不错的效果。

综上可得本课题的研究在自动问答方面已经取得了一定的成果，但其中同样存在着一些不足之处，主要有如下几个方面：

（1）语料集质量受到搜索引擎的制约。虽然在语料构建过程中使用搜狗各个行业的种子词作为种子保证了领域的多样性，但由于语料库的原始语料来自于搜索引擎的检索，导致了构造出来的复述对一定程度上依赖于搜索引擎的搜索效果，数据筛选和人工标注初步解决了这一问题，如果要彻底解决这一问题则需要大量

的人工标注。

（2）对问句的理解能力有待加强。由于目前人工智能相关技术所限，计算机对问句的理解大部分是基于浅层语义的，例如句子主题发现、文本分类，问答系统在一些情况下并不能理解用户提供的自然语言表述句子的真正含义，这也制约了系统的性能。

（3）语义匹配算法时间效率较低。对于深度学习相关的算法来说，时间效率是制约该算法在实际工程中使用的重要因素，本课题中提到的基于注意力机制的卷积神经网络算法同样面临着这一问题。

由于时间及人力所限，本课题研究过程中暂时只采用了一些简单的方法来缓解这些问题，在今后的科研过程中本人将继续改进这些问题使得自动问答系统取得更好的效果。

参考文献

[1] 吴友政, 赵军, 段湘煜, 等. 问答式检索技术及评测研究综述[J]. 中文信息学报, 2005, 19(3):1-13.

[2] 张志昌, 张宇, 刘挺, 等. 开放域问答技术研究进展[J]. 电子学报, 2009, 37(5):1058-1069.

[3] Green B, Wolf A, Chomsky C, et al. Baseball, an Automatic Question-Answerer[C]//International Workshop on Managing Requirements Knowledge. 1961: 219-224.

[4] Terry Winograd. Five Lectures on Artificial Intelligence [J]. Linguistic Structures Processing, 1997, 5: 399- 520.

[5] Woods W A. Lunar Rocks In Natural English: Explorations in Natural Language Question Answering [J]. Linguistic Structures Processing, 1977, 5: 521−569.

[6] Ferrucci D, Brown E, Chu-Carroll J, et al. Building Watson: An Overview of the DeepQA Project[J]. AI Magazine, 2010, 31(3): 59-79.

[7] Strickland E. Can an AI Get into the University of Tokyo?[J]. IEEE Spectrum, 2013, 50(9): 13-14.

[8] Turing A M. Computing Machinery and Intelligence[J]. Mind, 1950, 59(236): 433-460.

[9] Voorhees E M. The TREC-8 Question Answering Track Report[C]//TREC. 1999, 99: 77-82.

[10] 王宝勋, 刘秉权, 孙承杰, 等. 网络问答资源挖掘综述[J]. 智能计算机与应用, 2012, 2(6):54-58.

[11] Zhang D, Lee W S. Question Classification using Support Vector Machines[C]//Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval. ACM, 2003: 26-32.

[12] Li X, Roth D. Learning Question Classifiers[C]//Proceedings of the 19th international conference on Computational Linguistics. 2002, 1: 1-7.

[13] Cui H, Kan M Y, Chua T S. Unsupervised Learning of Soft Patterns for Generating Definitions from Online News[C]//Proceedings of the 13th international conference on World Wide Web. ACM, 2004: 90-99.

[14] Clarke C L A, Cormack G V, Lynam T R, et al. Question Answering by Passage Selection[M]//Advances in Open Domain Question Answering. 2008: 259-283.

[15] Ittycheriah A, Franz M, Zhu W J, et al. IBM's Statistical Question Answering System[J]. Experimental Techniques, 2006, 33(6):30-37(8).

[16] Lee G G, Seo J, Lee S, et al. SiteQ: Engineering High Performance QA System Using Lexico-Semantic Pattern Matching and Shallow NLP[C]//TREC. 2002: 442-451.

[17] Riezler S, Vasserman A, Tsochantaridis I, et al. Statistical Machine Translation for Query Expansion in Answer Retrieval[C]//Proceedings of the 2007 Annual Meeting of the Association For Computational Linguistics. 2007, 45(1): 464.

[18] Bengio Y, Schwenk H, Senécal J S, et al. Neural Probabilistic Language Models[J]. Journal of Machine Learning Research, 2001, 3(6):1137-1155.

[19] Kim Y. Convolutional Neural Networks for Sentence Classification[C]. //Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing. 2014: 1746–1751.

[20] Zhou X, Hu B, Chen Q, et al. Answer Sequence Learning with Neural Networks for Answer Selection in Community Question Answering[C]//Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics. 2015: 713-718.

[21] Shang L, Lu Z, Li H. Neural Responding Machine for Short-Text Conversation [C]//Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics. 2015: 824-830.

[22] Krizhevsky A, Sutskever I, Hinton G E. ImageNet Classification with Deep Convolutional Neural Networks[C]//Advances in Neural Information Processing Systems. 2012: 1097-1105.

[23] Mikolov T, Karafiát M, Burget L, et al. Recurrent Neural Network based Language Model[C]//INTERSPEECH. 2010, 2: 3.

[24] Kombrink S, Mikolov T, Karafiát M, et al. Recurrent Neural Network Based Language Modeling in Meeting Recognition[C]//INTERSPEECH. 2011, 11: 2877-2880.

[25] Kalchbrenner N, Grefenstette E, Blunsom P. A Convolutional Neural Network for Modelling Sentences[C]//Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics. 2014: 655-665.

[26] Hu B, Lu Z, Li H, et al. Convolutional Neural Network Architectures for Matching Natural Language Sentences[C]//Advances in Neural Information Processing Systems. 2014: 2042-2050.

[27] Silver D, Huang A, Maddison C J, et al. Mastering the Game of Go with Deep Neural Networks and Tree Search.[J]. Nature, 2016, 529(7587):484-489.

[28] Dolan B, Quirk C, Brockett C. Unsupervised Construction of Large Paraphrase Corpora: Exploiting Massively Parallel News Sources[C]//Proceedings of the 2004 Conference on Computational Linguistics. 2004: 350.

[29] Socher R, Huang E H, Pennington J, et al. Dynamic Pooling and Unfolding

Recursive Autoencoders for Paraphrase Detection[C]//Advances in Neural Information Processing Systems, 2011: 801-809.

[30] He H, Gimpel K, Lin J. Multi-perspective Sentence Similarity Modeling with Convolutional Neural Networks[C]//Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. 2015: 1576-1586.

[31] Potthast M, Stein B, Barrón-Cedeño A, et al. An Evaluation Framework for Plagiarism Detection[C]//Proceedings of the 23rd International Conference on Computational Linguistics. Association for Computational Linguistics, 2010: 997-1005.

[32] Ganitkevitch J, Vandurme B, Callison-Burch C. PPDB: The Paraphrase Database[C]//Proceedings of The 2013 Conference on the North American Chapter of the Association for Computational Linguistics. 2013: 758-764.

[33] Mikolov T, Yih W, Zweig G. Linguistic Regularities in Continuous Space Word Representations[C]//Proceedings of the 2013 Conference on the North American Chapter of the Association for Computational Linguistics. 2013: 746-751.

[34] Mikolov T, Sutskever I, Chen K, et al. Distributed Representations of Words and Phrases and their Compositionality[C]//Advances in Neural Information Processing Systems, 2013: 3111-3119.

[35] Mihalcea R, Tarau P. TextRank: Bringing Order into Texts[C]//Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing. 2004:404-411.

[36] Kusner M J, Sun Y, Kolkin N I, et al. From Word Embeddings to Document Distances[C]//Proceedings of the 32nd International Conference on Machine Learning. 2015: 957-966.

[37] Madnani N, Tetreault J, Chodorow M. Re-examining Machine Translation Metrics for Paraphrase Identification[C]//Proceedings of the 2012 Conference on the North American Chapter of the Association for Computational Linguistics. 2012:182-190.

[38] Ji Y, Eisenstein J. Discriminative Improvements to Distributional Sentence Similarity[C]//Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing. 2013: 891-896.

[39] LeCun Y, Bengio Y, Hinton G. Deep Learning[J]. Nature, 2015, 521(7553): 436-444.

[40] 刘一佳, 车万翔, 刘挺, 等. 基于序列标注的中文分词、词性标注模型比较分析[J]. 中文信息学报, 2013, 27(04): 30-36.

[41] Srivastava N, Hinton G, Krizhevsky A, et al. Dropout: a Simple Way to Prevent Neural Networks from Overfitting[J]. Journal of Machine Learning Research, 2014, 15(1):1929-1958.

[42] Larochelle H, Hinton G E. Learning to Combine Foveal Glimpses with a Third-order Boltzmann Machine[C]//Advances in Neural Information Processing Systems. 2010: 1243-1251.

[43] Bahdanau D, Cho K, Bengio Y. Neural Machine Translation by Jointly Learning to Align and Translate[J]. arXiv preprint arXiv:1409.0473, 2014.

[44] Sukhbaatar S, Weston J, Fergus R. End-to-end Memory Networks[C]//Advances in Neural Information Processing Systems. 2015: 2440-2448.

[45] Sutskever I, Vinyals O, Le Q V. Sequence to Sequence Learning with Neural Networks[C]//Advances in Neural Information Processing Systems, 2014: 3104-3112.

攻读学位期间发表的学术论文

哈尔滨工业大学学位论文原创性声明和使用权限

学位论文原创性声明

本人郑重声明：此处所提交的学位论文《基于CNN语义匹配的自动问答系统构建方法研究》，是本人在导师指导下，在哈尔滨工业大学攻读学位期间独立进行研究工作所取得的成果，且学位论文中除已标注引用文献的部分外不包含他人完成或已发表的研究成果。对本学位论文的研究工作做出重要贡献的个人和集体，均已在文中以明确方式注明。

作者签名：邓憧　　日期：2017 年 1 月 6 日

学位论文使用权限

学位论文是研究生在哈尔滨工业大学攻读学位期间完成的成果，知识产权归属哈尔滨工业大学。学位论文的使用权限如下：

（1）学校可以采用影印、缩印或其他复制手段保存研究生上交的学位论文，并向国家图书馆报送学位论文；（2）学校可以将学位论文部分或全部内容编入有关数据库进行检索和提供相应阅览服务；（3）研究生毕业后发表与此学位论文研究成果相关的学术论文和其他成果时，应征得导师同意，且第一署名单位为哈尔滨工业大学。

保密论文在保密期内遵守有关保密规定，解密后适用于此使用权限规定。

本人知悉学位论文的使用权限，并将遵守有关规定。

作者签名：邓憧　　日期：2017 年 1 月 6 日

导师签名：　　日期：2017 年 1 月 6 日

致　谢

在毕业论文即将交稿之际，本人两年半的研究生生涯也即将迎来尾声。在哈尔滨工业大学深圳研究生院就读的两年让我学到了很多知识技能，这些东西将令我受益终生，在此感谢所有帮助过我的老师、同学、亲人以及朋友。

感谢我的导师王晓龙教授，老师在学习上的细心指导以及生活上的无私关怀给予了我极大的支持和鼓励。在科研方面，王老师严谨的治学态度以及开放的科研精神潜移默化的影响着我，使我在平时的学术研究中养成了细致的习惯，培养了我独立思考的能力。王老师常常教育我们要学以致用，提倡在实际工程中对我们的科研成果进行实践，这种务实的科研精神将对我的生活和学习产生深远的影响。在课题的选择、进行以及论文的撰写方面，王老师也给予了我很大的帮助，使得我在课题完成的过程中获得了很大的收获。

感谢问答项目组的指导老师陈清财教授，在项目进行的过程中，陈老师的悉心指导对我的科研工作提供了很大的帮助。在定期的工作总结中，陈老师希望我们能有条理的对已完成的工作进行总结，并针对下一阶段的工作做出详细的计划，使得我的科研、学习效率都得到了很大的提高。对于我已完成的科研工作，陈老师也会做出客观的评价以及指导，培养了我细致、认真思考的能力。

感谢智能计算实验室的徐睿峰老师、丁宇新老师、刘滨老师、汤步洲老师，从论文开题到即将结稿他们给予了大量的指导和帮助，并且在实验室的日常生活中也给予了关心。感谢阿里巴巴云计算的曾华军老师、孙健老师以及其他 idst 的同事对本课题给予的支持和帮助。

感谢在本课题中指导我的博士刘欣师兄，他的帮助使我迅速的投入到了课题的研究中并且用亲身示例让我学到了扎实、耐心的科研精神。同时感谢项目组的技术顾问周小强、户保田师兄，两位博士在项目进行过程中提供的建议及指导推动了课题的研究进展，也使得我学到了更多的科研方面的知识。感谢相洋师兄在早期的科研工作中对我进行的指导和建议，使得我掌握了本课题研究中所涉及的基本方法。

感谢实验室的潘囿丞、唐朝红、步君昭、刘超、刘雨朦、庄烈斌、王远同学以及我的室友刘文强、胡可同学，无论是生活还是学习上，他们都对我产生了非常大的助力。感谢 13 级的洪清华、向鑫、林家欣师兄，在项目的进行过程中对我进行指引。

最后，感谢支持我的亲人以及朋友，你们的支持将是我日后成长和前行的不懈动力。