|  |  |
| --- | --- |
| Inline Text Wrapping Picture | Inline Text Wrapping Picture |

硕士研究生学位论文阶段报告

学 号: 2017140509

姓 名: 王瑾

学 院: 计算机学院

专业(领域): 计算机技术

研究方向: 大数据技术与智能信息处理

导师姓名:

北京邮电大学

2019年12月9日

|  |  |  |  |
| --- | --- | --- | --- |
| 论文题目 | 基于深度学习的多段落阅读理解系统的设计与实现 | | |
| 论文类型 | 应用研究 | 选题来源 | 其他 |
| 开题日期 | 2018-10-12 | 是否开题题目 | 否 |
| 论文开始日期 | 2018-10-12 | 报告日期 | 2019-12-11 |
| 报告地点 | 教三楼912 | 报告时间 | 上午 8:30 |
| |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | **一、研究内容简介**（包括：选题背景、研究内容、关键技术、论文计划、论文进度及目标，不少于：5000）  **1、选题背景**  随着大数据时代的到来，网络中非结构化文本迅速增加，海量非结构化文本已经成为我们获取有价值的信息的重要方式。因此这为更加精准的搜索引擎提出了挑战，而多段落阅读理解作为其中非常重要的一个环节，越来越多的成为研究的热点和重点。同时，教会机器理解人类的语言一直是自然语言处理领域的一项挑战。为解决自然语言处理中的各项任务，如词性标注，命名实体识别，句法解析，指代消解等，学术界和工业界进行了长期的研究。而自然语言理解可以测试所有这些任务，甚至完成对语义更深层的理解。正如我们通过阅读理解题目测试人对于文章的理解能力一样，机器阅读理解也可以测试机器对于自然语言的理解能力。  阅读理解任务是给定一组训练样本，目标是学习一个预测模型，这个模型把文章和相应的问题作为输入，输出是问题的答案,即[1]。根据问题对应的段落数量不同，可以把阅读理解任务分为单段落阅读理解和多段落阅读理解[2]。根据答案的形式不同，可以把阅读理解任务分为完形填空任务（cloze style），多项选择任务(multiple choice)，答案抽取任务(span prediction)和自由回答任务(free-form answer)。  随着深度神经网络的快速发展和数据集的迅速扩充，单段落抽取式阅读理解发展迅速。Shuohang Wang等人[3]搭建了通过问题表示段落的match-lstm模型，并使用Vinyals 等人[4]提出的pointer网络抽取答案。Seo等人[5]提出双向注意力流网络，并通过多个维度表示问题-段落对。Furu Wei等人[6]提出R-net阅读理解模型，模型首次在SQuAD1.0数据集上达到接近人工标注的效果。随着BERT[7]模型的提出，阅读理解任务的效果迅速提升，在很多数据集上效果已经接近甚至超过人工标注的效果。  在多段落阅读理解任务中，Lee等人[8],Lin等人[9]和Kratzwald 等人[10]提出首先检索出top-n的相关段落，然后再阅读理解，然而这个过程忽略了阅读理解和检索过程中存在的丰富信息。杨志明等人[11]提出基于多段落重排序的方法。Danqi Chen等人[12]提出了把命名实体、句法信息等加入阅读理解任务中，并提出基于TF-IDF和N-gram的搜索引擎算法。Shuohang Wang[13]等人提出基于强化学习的多段落阅读理解方法。Lee等人[8]提出根据答案和候选段落的分数对答案排序。Shuohang Wang等人[14]提出结合多种答案特征对候选答案排序的算法。Bernhard等人[15]提出融合信息检索特征和阅读理解特征的候选答案排序方法。然而这些方法并没有把问题、答案和段落之间的复杂语义信息考虑在内，模型效果都有待提高。  深度学习具有强大的非线性拟合能力，并能够实现从复杂的特征中提取有主要意义的特征，并且已经在自然语言处理等复杂问题方面取得了巨大的成功。同时经过学术界和工业界的长期研究，自然语言处理中的一些基本任务如句法依存关系，文本蕴含关系等已经取得了非常好的效果，而这些基本的句法关系对于机器理解文章也是至关重要的。因此本文针对多段落阅读理解中存在的问题，提出了融合文本蕴含，句法依存关系和深度学习的模型：基于BERT和句法依存关系注意力机制的阅读理解模型和融合文本蕴含关系特征和其它文本特征的候选答案排序模型。同时开发了基于深度学习的多段落阅读理解系统，系统包括两个模块，分别是阅读理解分析模块和开放域问答模块。  **2、研究内容**  本文提出使用基于句法依存关系注意力机制的阅读理解模型和融合文本蕴含特征和其它文本特征的候选答案排序模型，解决多段落阅读理解问题。针对阅读理解中文本长度过长，传统注意力机制会对所有词都计算注意力权重，导致一些噪声数据的干扰的问题。本文提出一种基于句法依存关系注意力机制的阅读理解模型。句法依存关系通过分析句子结构、句法，生成句法依存树。注意力机制会根据句法依存树给有依存关系的词赋予更高的权重，从而提高了长文本阅读理解效果。针对多段落阅读理解候选答案有多个，需要找出最佳答案的问题。本文提出基于文本蕴含关系特征和多种其它特征的候选答案排序模型。针对基于二分类模型进行候选答案排序效果差的问题，使用了基于候选答案对损失函数的候选答案排序模型。针对目前缺少可解释的中英文阅读理解系统，以及缺少基于多段落阅读理解的开放域问答系统的问题，本文设计开发了基于深度学习的多段落阅读理解系统，系统分为两个模块，分别是阅读理解分析模块和开放域问答模块。研究主要分为以下三个方面：  **1)针对长文本的阅读理解算法研究**  阅读理解任务是根据给定的段落和问题，从段落中抽取出问题的答案。通常采用基于注意力机制的模型，该类模型一般分为三层，问题编码层，文章编码层和答案预测层。为充分表示问题和文章的相关关系，在问题编码层和文章编码层会引入注意力机制，如match-lstm[3], Attention over Attention [15], BiDAF [5], DRQA [11]等模型。基于BERT的阅读理解模型出现之后，基于BERT的注意力机制计算方法是问题和文章拼接后进行整体注意力权重计算。然而这些模型都是对所有词计算注意力权重，会引起很多噪声干扰，并且随着文本长度的增加，影响会越来越严重。  针对以上问题，文本提出一种基于句法依存关系注意力机制的阅读理解模型，句法依存关系是一个树形结构，针对每个词本文只对这个词的所有祖先节点词计算注意力权重，从而有效的降低了噪声的干扰，提升了长文本阅读理解的效果。  **2)多段落阅读理解候选答案排序算法研究**  多段落阅读理解候选答案排序任务是对阅读理解后产生的多个候选答案进行排序，然后模型需要从候选答案中找到一个最佳答案，作为模型的最终输出。目前候选答案排序方法可分为以下四种：一是仅根据答案得分对候选答案排序，把得分最高的答案作为最佳答案。二是根据候选答案得分和候选答案出自的段落和问题的相似度两个特征对候选答案进行排序，把得分最高的候选答案作为最佳答案。三是采用全局归一化的方法，在模型最后一层把所有段落的向量表示连接起来，直接确定最佳答案。四是在使用答案得分，段落相似度的基础上，引入问题类型，答案词性，答案实体类型等特征，进行答案排序，并最终确定最佳答案。从根本上分析，以上各方法仅仅利用了问题、文章、答案的一些基本特征进行候选答案排序，并没有充分考虑三者之间的关系。本文通过分析大量数据，发现正确答案出自的句子会蕴含更多的问题和答案信息，而错误答案出自的句子虽然在句法描述上有很强的迷惑性，但是句子并不能蕴含所有的问题和答案信息。  针对以上问题，本文提出一种融合文本蕴含关系特征和其它文本特征的候选答案排序模型。本文把答案出自的句子作为前提，答案和问题连接作为假设，如果根据前提能够推理出假设，则认为两者之间存在蕴含关系。模型训练时，采用基于候选答案对的二分类训练方法，使用时，本文使用蕴含分数作为候选答案排序的一维特征，并结合其它如文本检索特征，阅读理解特征等进行候选答案排序。  **3)基于深度学习的多段落阅读理解系统研究**  多段落阅读理解模型不仅可以用于测试很多阅读理解任务，证明机器的理解能力。还可以应用于问答系统。本文整合上文提出的多段落阅读理解模型，并通过爬虫扩充开放域问答系统的数据量，设计实现基于开放域的问答系统[16][17]。同时针对深度学习模型可解释性差的问题，本文实现了可视化的阅读理解分析系统，可以对问题和段落的权重等进行可视化分析，从而增强用户对模型的理解。  **3、关键技术**  **1) 基于BERT和句法依存关系注意力机制的阅读理解模型**  随着非结构化文本数据量的增加，阅读理解任务中问题相关段落的长度也在不断增加，而传统的注意力机制如BiDAF, Stanford attention reader，self-attention[7]等对所有词都计算注意力权重，从而引起很多噪声干扰，影响阅读理解模型的效果，也增加了计算量。  考虑到通常情况下文本中的每个词与和这个词有直接或间接句法依存关系的词有较大的联系，而句法依存关系作为自然语言处理中的一项基本任务，已取得了很好的效果，并且stanfordcorenlp提供了句法依存关系接口，可以快速对中英文数据生成句法依存树。因此本文提出了基于BERT和句法依存关系注意力机制的阅读理解模型。模型的整体结构如图1所示，模型分为三部分：    图1 基于BERT和句法依存关系注意  力机制的阅读理解模型  **基于BERT 的编码层：**  模型的输入为问题和文章，上句是问题，下句是文章，并用[SEP]分隔符进行分割。  整个BERT模型采用多头注意力机制，把输入分为n个头，并采用position embedding的方法保证各个头的相对位置不变。对于每个头, 首先由输入向量分别点乘，，形成三个向量，，，   |  |  |  | | --- | --- | --- | |  | (1) | | |  | (2) | |  | (3) |   然后根据，，计算 attention向量   |  |  | | --- | --- | |  | (4) |   其中为向量维度，公式表示查询向量与键值向量进行点成并作归一化之后，经过函数，再与相乘，得到attention向量，然后把所有结果连接起来得到当前层的向量表示。   |  |  | | --- | --- | |  | (5) |   其中 *=* ，这样就完成了一层的运算，如图2所示，同时在不同层，即每个编码器中的每个子层都有一个残差连接，同时还伴随着一个规范化步骤。    图2 每个头的self-attention计算过程  **基于句法依存关系注意力机制的向量表示层:**  首先分别将BERT最后一层第i个头的输出key, value和query分别表示为, , 则注意力矩阵可以表示为：   |  |  | | --- | --- | |  | (6) |   其中M为句法依存注意力矩阵，由于句法依存树是一个树形结构，本文假定每个词的所有祖先节点都与这个词有关联，即注意力机制只对有关联的词计算权重和：   |  |  | | --- | --- | |  | (7) |   第i个头基于句法依存关系注意力机制的向量表示为：   |  |  | | --- | --- | |  | (8) |   把所有头的连接起来然后过一层全连接层，然后过GeLU[7]激活函数，最后过一层归一化层，得到最终的向量表示为：   |  |  | | --- | --- | |  | (9) |   把BERT的原始输出和过语义依存关系注意力机制后的输出加和，作为最终的向量表示：   |  |  | | --- | --- | |  | (10) |   其中，  **答案抽取层:**  把上文最终的输出向量过全连接层和softmax层，得到每个词是开始和结束的概率。   |  |  | | --- | --- | |  | (11) |   损失函数采用交叉熵损失函数：   |  |  | | --- | --- | |  | (12) |   在预测时，取开始位置和结束位置分数和最大的范围作为最终答案，如果没有答案则答案范围为[0,0]。   |  |  | | --- | --- | |  | (13) |   **2) 基于候选答案对的融合蕴含关系特征和其它文本特征的候选答案排序模型**  由于在多段落阅读理解中，基于抽取的机器阅读理解模型对于存在误导的对抗性样本的表现非常脆弱，因此分数最高的答案不一定是最佳答案，所以需要综合其它特征，对候选答案排序。本文选取的特征有三类[14]，分别是文本检索特征，包括问题和段落的相似度，问题和段落的长度，问题的类型。阅读理解特征，包括答案的分数，包括开始位置分数，结束位置分数和总的分数，答案的实体类型和答案的词性。蕴含关系特征，蕴含句子对(问题和答案,答案出自的句子)的得分。  通过观察大量数据，发现正确答案出自的句子蕴含了更多的问题信息，因此本文提出基于蕴含关系特征的候选答案排序算法。本文采用基于BERT的句子对分类模型作为本文蕴含模型，模型如图3所示：    图3 基于BERT的句子对分类模型作为文本蕴含模型  除文本蕴含特征外，本文还选用另外两种特征即信息检索特征：如问题和文章的相似度、长度信息、问题类型等，阅读理解特征：如候选答案得分、答案词性和实体类型等。  提取完所有特征后，需要进行特征融合，由于从不同段落中抽取的答案有相同的情况，因此需要对相同答案的特征进行特征融合，融合方法为对相同答案的特征值取最大值、最小值、平均值和和值。具体抽取的特征和特征融合的方法如表1所示：  表1 抽取的特征和特征融合的方法    特征提取和特征融合后，假定每个候选答案的特征用（其中对于每一维度的特征本文做了minmax归一化处理）。然后过全连接层和ReLU激活函数，最终每个候选答案的得分为：   |  |  | | --- | --- | |  | (14) |   通过实验发现，如果把候选答案的得分过softmax层进行二分类，无论使用深层还是浅层的神经网络，效果都不理想。因此文本使用基于候选答案对的损失函数[19]，把候选答案之间的对比关系引入损失函数中：   |  |  |  | | --- | --- | --- | |  | (15) | | |  | (16) | |  | (17) |     其中代表sigmoid激活函数，代表对应的候选答案是否为正确答案，是为1，不是为0。  预测时，根据对所有候选答案排序，并把得分最高的答案作为正确答案。其中候选答案最多取10个。  **3) 基于深度学习的多段落阅读理解系统**  融合以上提出的两种算法，本文搭建了基于深度学习的多段落阅读理解系统，系统分为两个模块，分别是基于多段落阅读理解的中英文开放域问答模块和阅读理解分析模块。使用TensorFlow框架训练模型，使用GPU加速，通过爬虫扩充中文数据集，并构建elasticsearch索引用于文档检索。在web层，使用flask实现python接口，并使用css，vue和js实现前端编程。  整个系统的处理流程分为模型离线训练和实时分析两个部分。在实时分析部分，会根据任务的不同调用不同的接口和模型，对于阅读理解分析模块，接口会返回问题的答案，attention矩阵，段落和问题中的重要词语。对于基于多段落阅读理解的中英文开放域问答模块，首先会以对话框的形式返回问题的答案，同时还可以查询答案对应的段落以及所有候选答案和候选段落。  **4、论文计划**  表2 论文计划   |  |  | | --- | --- | | 2018.09-2018.12 | 前期调研，确定研究方向与目标, 制订实施方案。 | | 2019.01-2019.03 | 学习Tensorflow框架，掌握TensorFlow架构及工作原理，熟悉TensorFlow的使用，可以搭建模型 | | 2019.04-2019.07 | 数据分析处理和搭建模型，完成整个模型的搭建 | | 2019.08-2019.11 | 优化模型，确定最终模型 | | 2019.12-2020.1 | 系统实现，并发表一个专利 | | 2020.02-2020.03 | 系统各模块整合测试，提升系统性能 | | 2020.04-2020.05 | 完成论文撰写，完成答辩材料的准备 |   **5、论文进度及目标**  截止目前，前五项任务已全部完成，第六项任务完成80%，完成开题计划中应该完成的工作。  本文旨在使用深度学习和传统自然语言处理任务相结合的方法，完成多段落阅读理解算法的实现和系统的设计开发。针对长文本阅读理解中每个词对所有词都计算注意力权重，从而导致对一些无用词的注意，引起噪声干扰的问题，本文利用句法依存关系对句子结构强大的分析能力，提出了句法依存关系注意力机制，同时由于BERT在众多自然语言处理任务中都表现出很强的效果，因此结合这两点本文提出了基于BERT和句法依存关系的注意力机制的阅读理解模型。针对多段落阅读理解中候选答案有多个，而只根据答案得数或者答案和段落相似度分数进行排序的方法效果不好的问题，本文提出基于蕴含关系特征和其它文本特征的候选答案排序算法，并针对基于二分类的方法排序效果不好的问题，提出了基于句子对损失函数的候选答案排序算法。结合以上两种算法，本文设计实现了基于深度学习的多段落阅读理解系统，系统分为两个模块，分别是阅读理解分析模块和开放域问答模块。  论文整体架构及内容如下：  第一章为绪论，主要介绍本论文的研究背景及意义，并通过阅读大量文献归纳国内外研究发展现状，进而提出本论文的研究内容和和组织架构。  第二章为详细介绍阅读理解算法，包括不同的阅读理解任务和算法，评价指标等。  第三章为相关技术介绍，主要介绍本论文需要用到的相关技术，包括基于TensorFlow的组件研究、阅读理解算法的研究、句法依存关系的研究，文本蕴含关系的研究，BERT模型的研究，候选答案排序算法的研究。  第四章为基于BERT和句法依存关系注意力机制的阅读理解模型研究，以及实验设置和模型效果展示。  第五章为融合文本蕴含特征和其它文本特征的候选答案排序算法研究，以及实验设置和模型效果展示。  第六章为基于深度学习的多段落阅读理解系统，包括系统需求分析，数据收集方式，各模块设计和系统展示。  第七章为总结与展望，客观地总结在论文的写作、算法实现和系统设计方面主要做的工作，并给出论文的写作、算法实现和系统设计和测试环节中发现的不足之处，并对以后的改进进行展望。  目前，本文已完成对基于深度学习的多段落阅读理解算法的调研，模型搭建和调优，实验结果分析和系统设计与实现部分。  接下来的时间里，将进一步优化系统；预计在2019年3月前完成论文初稿。并广泛征求老师和同学的建议，对论文进行修改，使论文的整体结构更加合理，论述更加专业化，在2019年5月前完成论文的修改工作，产出最终的毕业论文，并完成毕业答辩各项准备工作。  **二、论文进展情况**（包括：报告工作计划、实际进展情况）  **1、工作计划**  本文旨在：研究长文本下的阅读理解模型和多段落阅读理解中候选答案排序模型。对于长文本阅读理解问题，本文提出利用句法依存关系的注意力机制，解决噪声干扰问题。对于多段落阅读理解中候选答案的排序问题，本文提出基于文本蕴含特征和其它文本特征的候选答案排序模型，充分考虑了问题，答案和文章三者之间的关系。融合以上两种算法，本文实现了基于深度学习的多段落阅读理解系统，系统分为阅读理解分析模块和中英文开放域问答模块。具体工作计划和安排如表3所示：  表3具体工作计划和安排   |  |  |  | | --- | --- | --- | | **时间** | **工作内容** | **预期成果** | | 2018.09-2018.12 | 前期调研，确定研究方向与目标, 制订实施方案。 | 了解现有方法的不如，初步确定解决方案 | | 2019.01-2019.03 | 学习Tensorflow框架，掌握TensorFlow架构及工作原理，熟悉TensorFlow的使用，可以搭建模型。 | 能熟练使用TensorFlow框架 | | 2019.04-2019.07 | 数据分析处理和搭建模型，完成整个模型的搭建和调优 | 完成模型编码，并发表一篇小论文 | | 2019.08-2019.11 | 优化模型 | 优化模型，完成专利撰写 | | 2019.12-2020.1 | 搭建系统 | 提交一篇专利 | | 2020.02-2020.03 | 系统各模块整合测试，提升系统性能 | 完成系统测试 | | 2020.04-2020.05 | 论文终稿撰写，完成答辩材料的准备 | 完成论文终稿和答辩材料 |   **2、实际进展情况**  目前，实际进展情况如表4所示：  表4 实际进展情况   |  |  |  | | --- | --- | --- | | **时间** | **工作成果** | **完成情况** | | 2018.09-2018.12 | 了解了现有方法的不足，初步确定了解决方案 | **已完成** | | 2019.01-2019.03 | 熟练使用TensorFlow框架 | **已完成** | | 2019.04-2019.07 | 完成模型编码，并发表一篇小论文 | **已完成** | | 2019.08-2019.11 | 优化模型，完成专利撰写 | **已完成** | | 2019.12-2020.1 | 设计实现基于深度学习的多段落阅读理解系统，并提交一篇专利 | **已完成** | | 2020.02-2020.03 | 完成系统测试和优化 | 未完成 | | 2020.04-2020.05 | 完成论文终稿和答辩材料 | 未完成 | | | | | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | **三、工作成果**（目前已完成学位论文工作的内容、取得的阶段性成果和主要创新点。包括所完成的理论和实验研究以及所获得的结论；已发表的与学位论文相关的学术论文等,字数不少于：1500)  **1、已完成学位论文工作的内容**  本文旨在使用深度学习和传统自然语言处理任务相结合的方法对多文档做阅读理解。本文对BERT模型，句法依存关系，注意力机制进行深入研究，提出基于BERT和句法依存关系注意力机制的阅读理解模型，并实现了基于Tensorflow的模型搭建。为解决候选答案排序问题，本文对候选答案排序算法，文本蕴涵关系进行了深入研究，提出基于文本蕴含关系和多种特征的候选答案排序算法，同时为了解决基于传统二分类方法导致分类效果差的问题，本文提出了基于候选答案对的候选答案排序算法。为了提高阅读理解模型的可分析性，同时将本文提出阅读理解算法和候选答案排序算法应用到实际中，本文设计开发了基于深度学习的多段落阅读理解系统。系统分为离线训练和实时预测两个部分，离线训练部分包括数据集的处理、基于BERT和句法依存关系注意力机制的阅读理解模型训练、基于文本蕴含关系和多种特征的候选答案排序模型训练、对模型进行验证、模型保存。在线预测部分包括，对于问答平台，首先输入问题，通过ES检索相关文档，并找出top 5的段落，加载模型，根据模型进行答案预测。对于阅读理解分析平台，会针对输入的问题和文档，预测问题的答案，同时输出注意力矩阵和问题和文档中权重最高的五个词语。  **已经完成的工作内容如下：**  ①环境搭建，数据处理如表5所示：  表5环境配置和选取的数据集   |  |  | | --- | --- | | 环境 | 版本号 | | TensorFlow | 1.12.0 | | CUDA | 9.0 | | GPU | 1080Ti，11G | | 数据集 | 语言 | | SQuAD2.0 | 英文 | | DuReader | 中文 | | DRCD | 中文 | | WebQuestions | 英文 | | Quasar-T | 英文 | | WikiMovies | 英文 | | SQuADopen | 英文 | | CuratedTREC | 英文 |   ②完成注意力机制、BERT算法、句法依存关系、文本蕴含关系的算法研究。  ③完成基于BERT和句法依存关系注意力机制的阅读理解模型的搭建和训练。  ④完成融合文本蕴含特征和其它文本特征的候选答案排序模型的搭建和训练。  ⑤使用css，vue，js进行web端界面实现，并使用flask实现python接口封装。实现基于深度学习的多段落阅读理解系统。  **2、取得的阶段性成果**  **1）数据处理和评估方法研究**  **①数据集描述和处理：**  本文采用SQuAD2.0[21][22], DuReader[23]和DRCD[24]三个数据集对基于BERT和句法依存关系注意力机制的阅读理解模型进行评估。  SQuAD2.0[22]是斯坦福提出的英文阅读理解数据集，数据集统计结果如表6所示。答案都给出了开始位置和结束位置，并且每个问题只有一个对应段落。对于没有答案的问题，数据集还给出了可疑答案。  表6 SQuAD2.0数据集统计结果   |  |  |  | | --- | --- | --- | |  | all | no-answer | | train | 130,319 | 43,498 | | dev | 11,873 | 5,945 | | test | 8,862 | 4,332 |   DuReader[23]是百度提出的中文多段落阅读理解数据集，数据集统计结果如表7所示。其中各个类型的问题和答案的比例如表8所示。在阅读理解实验中，文本只用到其中的golden paragraph和golden answer两个值。  表7 DuReader数据集统计结果   |  |  |  | | --- | --- | --- | | train | dev | test | | 271,574 | 10,000 | 20,000 |   表8 DuReader数据集各个类型的问题和答案的比例    DRCD[24]是一个中文的阅读理解数据集，数据形式和SQuAD1.0类似，即所有问题都是可回答的。数据集统计结果如表9所示。答案都给出了在文中的开始位置和结束位置，并且每个问题只对应一个段落。但是每个段落可能对应多个问题。  表9 DRCD数据集统计结果   |  |  |  | | --- | --- | --- | | train | dev | test | | 26,932 | 3,524 | 3,485 |   三个数据集的对比结果如图4,5所示：    图4 各数据集训练集测试集和验证集数据量对比    图5 各数据集段落平均长度对比  本文采用WebQuestions[25],Quasar-T[26], WikiMovies[27], SQuADopen[21][22]和CuratedTREC[28]五种数据集对基于候选答案对的融合文本蕴含特征和其它文本特征的候选答案排序算法进行验证，数据集对比结果如表10，图6所示。除去Quasa-T和SQuAD数据集外，WebQuestions, WikiMovies和CuratedTREC数据集都没有提供候选段落，因此本文使用[12]对数据集处理后的结果进行试验。处理后的数据集已经进行了文档检索，每个问题有对应的五个分数最高的段落，可以直接用于本文阅读理解和候选答案排序任务。  表10 各数据集训练集和测试集的数量   |  |  |  | | --- | --- | --- | |  | Train | Test | | Quasar-T | 28,496 | 3,000 | | SQuADopen | 71,231 | 10,570 | | WikiMovies | 36,301 | 9,952 | | CuratedTREC | 3,464 | 694 | | WebQuestions | 4,602 | 2,032 |     图6 各数据集训练集和测试集数据量对比  WebQuestions[25]是构建的问答数据集，是从Freebase KB中获取的数据，本文使用[12]对数据进行处理后的版本，即把Freebase-IDs替换成答案。  WikiMovies[27]数据集包含来自电影行业的数千个问答对。它的设计使得所有问题都可以由知识库（即OpenMovie Database）或全文内容（Wikipedia）来回答。  CuratedTREC[28]数据集是来自文本检索会议（TREC）的问答任务数据集。  SQuADopen[21][22]数据集是斯坦福的问答数据集，包含100000个问题-答案-段落三元组，本文用SQuADopen数据集，即忽略了段落信息。  Quasar-T[26]数据集由问答对构成，并通过搜索引擎从ClueWeb09中抓取相关段落。  **②评估方法选择：**阅读理解测评方法可分为两类，分别是针对抽取式阅读理解的评估方法和针对自由回答形式的阅读理解的评估方法。  针对抽取式的阅读理解评估方法有两种，分别是EM(Exact match)和F1 score。  EM评估方法：如果预测答案和正确答案完全相同则为1，否则为0。  F1 score评估方法：计算预测答案和正确答案的平均词语重合程度。预测答案和正确答案都以词袋的方式表示，词级别的F1 score计算公式为：   |  |  | | --- | --- | |  | (22) | |  | (23) | |  | (24) |   针对自由回答形式的阅读理解任务，评估方法类似机器翻译和摘要生成算法的评估方法，分别为计算ROUGE和BLEU两个指标。  ROUGE(Recall-Oriented Understudy for Gisting Evaluation)是一种基于召回率的相似性度量方法，ROUGE包含四种方法，GOUGE-N,ROUGE-L,ROUGE-W,ROUGE-S，对于阅读理解算法的评估，我们主要是用ROUGE-L评估方法，L即是LCS(longest common subsequence),最长公共子序列，ROUGE-L的计算公式为：   |  |  |  | | --- | --- | --- | |  | (25) | | |  | (26) | |  | (27) |   其中表示正确答案和预测答案的最长公共子序列长度，m,n分别表示正确答案和预测答案的长度。  BLEU(Bilingual Evaluation Understudy) BLEU 采用一种N-gram的匹配规则，计算公式为：   |  |  | | --- | --- | |  | (28) |   其中表示单词k(n-gram)在候选答案中出现的次数，表示单词k(n-gram)在正确答案中出现的最多次数，这是标准答案有多个情况下的计算公式。  为了防止随着候选答案长度变短而效果变好的情况,BLEU在最后的平分结果中引入了长度惩罚因子:   |  |  | | --- | --- | |  | (29) |   其中代表候选答案的长度，代表正确答案的长度。  最终的评价公式为：   |  |  | | --- | --- | |  | (30) |   BLEU的原型系统采用的是均匀加权，即。N的上限取值为4。  **2）基于BERT和句法依存关系注意力机制的阅读理解模型实验结果**  本文实现了基于TensorFlow的阅读理解模型，采用stanfordcorenlp接口获取句法依存关系树。采用长度为200，步长为150的滑动窗口进行阅读理解。在SQuAD2.0,DuReader和DRCD三个数据集上进行模型测试。针对SQuAD2.0和DRCD数据集，评测标准为EM和F1。针对DuReader数据集match-lstm模型和BiDAF模型的测评标准为BLEU和Rouge-L，BERT finetune模型和本文使用的模型的测评标准为EM和F1。针对SQuAD2.0,如果答案的开始位置和结束位置都为0，则认为答案不存在。针对DuReader数据集，本文仅使用golden paragraph和golden answer进行测评。模型结果如表11所示：  表11实验结果    实验结果显示，本文提出的模型在三个数据集上的效果均高于BERT finetune baseline模型。其中在DuReader数据集上的效果最明显，分析原因可能是因为DuReader数据集的平均长度较长，图7为各个数据集文章长度的对比。虽然DRCD的段落长度也很长，但是由于数据集简单，并且模型效果已经接近人工标注的效果，所以效果提升不明显。    图7 各数据集段落平均长度对比  通过横向对比，DuReader数据集的效果最差，是因为百度阅读理解数据集的答案类型比较复杂，图8为各种类型答案所占的比例，任务难度本身就比较大。    图8 DuReader数据集各种类型答案所占的比例  而SQuAD2.0数据集由于存在没有答案的问题，图9为有答案和没有答案的问题占的比例，所以导致模型效果没有达到最优。    图9 SQuAD2.0数据集有答案和没有答案的问题占的比例  **3）基于候选答案对的融合文本蕴含特征和其它文本特征的候选答案排序算法效果**  本文采用基于BERT句子对分类模型，构建基于（答案出自的句子，问题+答案）对的蕴含关系模型，将模型得到的分数作为一维特征。训练过程为，根据候选答案中的错误答案构建负样本，对正确答案采用远距离监督的方法构建正样本。在模型测试的过程中，为了便于与RankQA模型比较，本文首先使用RankQA模型使用的阅读理解模型就行测试。然后将阅读理解模型换成本文提出的基于BERT和句法依存关系注意力机制的阅读理解模型进行测试，测试结果如表12所示：  表12 实验结果，评估指标为EM    通过分析结果，本文提出的候选答案排序算法在WebQuestion，Quasar-T，WikiMovie和SQuADopen数据集上的效果优于RankQA模型。使用本文提出的阅读理解模型，在WebQuestion，Quasar-T和SQuADopen数据集中取得了几个对比模型中最好的效果。为了把模型应用到开放域的问答系统中，本文把所有数据集组合到一起进行模型验证，模型效果虽然低于最好的效果，但是效果相差不大，所以本模型是可以应用到开放域问答模块中的。  同时本文统计了只根据候选答案分数进行排序结果正确，而用本文提出的排序方法进行排序结果错误的数量，占所有采用本文排序方法结果错误的比例，统计结果如图10所示：    图10 模型效果对比  为了分析排序所使用特征的重要程度，本实验统计了针对每个数据集对结果影响最大的特征，统计结果如表13所示：  表13对候选答案排序影响最大的特征，评估指标为EM    通过分析，WebQuestion和WikiMovies数据集对文本蕴涵关系特征最敏感，而Quasar-T和CuratedTREC数据集对于候选答案分数特征最敏感，SQuADopen数据集对问题和段落的相似度最敏感。  文本同时对中文数据集做了测试，本文与百度阅读理解leadboard排行榜前九名对比，本文提出的算法效果达到了前九名的平均效果，结果如图11所示：    图11 DuReader数据集测试结果  **4）基于深度学习的多段落阅读理解系统**  结合上文提出的两种技术，本系统主要实现以下两个大的模块：一、阅读理解分析模块，由于深度学习模型的可解释性较差，所以需要对如attention矩阵，问题和文章中词的重要程度进行可视化分析，因此本文实现了阅读理解分析模块。二、多段落阅读理解可以应用在开放域的问答系统中，为了充分验证模型的效果和实用性，本文基于多段落阅读理解模型，实现了中英文开放域问答模块。  **①系统整体架构设计**  本文在结合网络爬虫数据和现有数据集的基础上，基于上文提出的两种模型，构建了一个基于深度学习的多段落阅读理解系统。系统整体架构如图12所示：    图12 系统架构图  系统自底向上主要包括基础支撑层，数据处理层，算法层和web展示层。  第一层是基础支撑层，包括GPU用于加速计算，TensorFlow框架用于搭建深度学习模型，scrapy爬虫用于扩充数据，本文主要抓取了百度知道的数据，Elasticsearch用于根据问题搜索相关文档。第二层是数据处理层，包括对数据获取，即检索到相关文档。候选段落获取，即获得用于回答问题的候选段落。以及数据预处理工作等。第三层是算法层，主要包括上文提出的两种算法和搜索引擎算法。第四层是web展示层，采用flask实现Python接口，用css，vue和js实现前端编程。  **②系统处理流程**  系统的整体处理流程如图13所示：    图13 系统流程图  系统分为模型离线训练和实时分析两个部分。在实时分析部分，会根据任务的不同调用不同的接口和模型，对于阅读理解分析模块，接口会返回问题的答案，attention矩阵，段落和问题中的重要词语。对于开放域问答模块，首先根据问题搜索相关段落，然后基于上文提出的多段落阅读理解算法生成最佳答案，并以对话框的形式返回问题的答案，同时还可以查询答案对应的段落和候选答案。  **③系统展示**  本文设计实现了基于深度学习的多段落阅读理解系统。系统分为两个大的模块，分别是阅读理解分析模块和开放域问答平台模块。系统界面如图14,图15所示：        图 14阅读理解分析平台      图15开放域问答平台  **5. 已发表的与学位论文相关的学术论文等**  1、**Wang Jin**, Gao Yan, Li Jie, Yu Yanhua, Li Binyang. 2019. 3R: Reading - Ranking - recognizing for multi-passage reading comprehension. In *Proceedings of 2019 IEEE 8th Joint International Information Technology and Artificial Intelligence Conference, ITAIC 2019*. **(EI检索)**  2、提交一篇专利《一种融合文本蕴含和多种特征的多段落阅读理解候选答案排序方法》  **6.主要创新点**  **1)实现基于BERT和句法依存关系注意力机制的阅读理解模型**  本文将自然语言处理中的基本任务句法依存关系应用到阅读理解中，结合句法依存树构建注意力矩阵，每个词只和这个词所有的祖先节点有注意力权重。传统的阅读理解中的注意力方法是对所有的词计算注意力权重，即对所有词都有注意力，而结合实际的语言习惯，在一段话中，某个词并非和所有词都有关系，反而如果和所有词都计算注意力权重会引起噪声的干扰，影响模型的效果。  本文提出基于BERT和句法依存关系注意力机制的阅读理解模型，模型在获得BERT最后一层输出的基础上，又加了一层类似transformer层，而在注意力权重矩阵计算时，根据句法依存关系注意力矩阵计算注意力权重。同时加入全连接和normalization。最终将BERT原始输出和过句法依存关系注意力层的输出加和，作为模型编码层的最终输出。  本文提出的方法在中文数据集DuReader, DRCD和英文数据集SQuAD2.0数据集上均取得了优于bert finetune baseline模型的效果。  **2)实现融合文本蕴涵特征和其它文本特征的候选答案排序方法**  为解决多段落阅读理解会产生不止一个答案的问题，采取的方法包括：一、找一个最佳段落进行阅读理解，但是这种方法对于最佳段落选择算法的要求很高。二、对每个段落进行阅读理解，选取候选答案中得分最高的答案最为最佳答案。这种方法忽略问题、段落和答案中丰富的语义信息。三、融合一些简单特征如问题和段落的相似度，答案分数等进行候选答案排序。这种方法在很多数据集中取得了优于前两种方法的效果，但是没有把问题、段落和答案三者的语义信息综合考虑。  因此本文提出基于文本蕴含和多种特征的候选答案排序方法，通过观察数据发现，正确答案出自的句子蕴含了更多的问题和答案信息，所以蕴含关系对为（问题+答案，答案出自的句子），本文采用基于BERT的句子对分类模型作为文本蕴含模型，用最终的分数作为文本蕴含分数。  通过实验发现，传统的基于二分类的方法在候选答案排序上的效果不好，因此本文使用了基于候选答案对损失函数的候选答案排序算法，模型在多个测试集上效果优于所有对照模型。  **3)基于多段落阅读理解的中英文开放域问答平台和阅读理解分析使用平台**  虽然学术界对阅读理解和问答的研究已非常多，但是目前缺少基于开放域多段落阅读理解的中英文问答系统，以及可视化的阅读理解分析系统。本文基于上文提出的两种算法，并且结合现有的搜索引擎算法，设计实现了基于深度学习的多段落阅读理解系统。同时，为了使阅读理解结果有更好的可解释性，本文设计实现了针对单段落的阅读理解分析模块，该模块不仅可以根据输入的问题和段落抽取答案，同时把注意力权重矩阵进行可视化展示，并且能够分析问题和段落中最重要的词语有哪些。 | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **四、下一步工作计划及进度安排**（内容、时间及工作量估计，字数不少于：150）  预计到2020年1月完成优化问答模块的相关段落搜索算法；到2020年3月中旬完成系统优化，提高系统性能。到2020年4月中旬完成论文初稿的撰写；到2020年5月中旬完成优化论文细节，准备完答辩所需材料。  表14 下一步计划及进度安排   |  |  |  | | --- | --- | --- | | 时间 | 研究内容 | 预期效果 | | 2019.12-2020.01 | 优化问答模块的相关段落搜索算法 | 在各个数据集上进行验证，提供返回的段落中包含正确答案比例 | | 2020.02-2020.03 | 进一步优化系统，提高系统性能，优化UI界面 | 完成系统搭建所有工作 | | 2020.03-2020.04 | 完成论文初稿撰写 | 撰写至少3万字的论文 | | 2020.04-2020.05 | 优化论文细节，准备答辩所需材料 | 完成论文，准备好答辩材料 | |
| **五、论文及整改方案**（论文后期工作存在的困难、问题及整改方案，字数不少于：200）  **1、论文后期工作存在的困难和问题**  1）通过实验发现，问答系统采用的相关段落搜索算法效果有待提高，需要提出更好的搜索和段落抽取算法。  2）系统的界面设计需要进一步优化，系统的功能需要进一步完善，系统的性能需要进一步提高。  **2、整改方案**  1）通过查阅相关文献，优化段落搜索算法和段落抽取算法。并通过实验对不同方法进行验证。  2）优化系统的功能，美化系统界面，提升系统响应速度。增加如用户登录，错误报告等功能，提升系统的性能和用户体验。 |

|  |
| --- |
| **参考文献**  [1] Danqi Chen. Neural Reading Comprehension and Beyond: [D]. Stanford: University of Stanford，2018.  [2] Christopher Clark and Matt Gardner. 2018. Simple and effective multi-paragraph reading comprehension. In *Association for Computational Linguistics (ACL)*, volume 1, pages 845–855.  [3] Shuohang Wang and Jing Jiang. 2016. Machine comprehension using match-lstm and answer pointer. In *Annual Conference of the North American Chapter of the Association for Computational Linguistics(NAACL)*.  [4] Oriol Vinyals, Meire Fortunato and Navdeep Jaitly. 2015. Pointer networks.In *Annual Conference on Neural Information Processing* *Systems(NIPS)*, pages 2692–2700.  [5] Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. 2017. Bidirectional attention ﬂow for machine comprehension.In *International Conference on Learning Representations (ICLR).*  [6] Wenhui Wang, Nan Yang, Furu Wei, Baobao Chang, and Ming Zhou. 2017. Gated selfmatching networks for reading comprehension and question answering. In *Association for Computational Linguistics (ACL)*, volume 1, pages 189–198.  [7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*.  [8] Jinhyuk Lee, Seongjun Yun, Hyunjae Kim, Miyoung Ko, and Jaewoo Kang. 2018. Ranking Paragraphs for Improving Answer Recall in Open-Domain Question Answering. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 565– 569.  [9] Yankai Lin, Haozhe Ji, Zhiyuan Liu, and Maosong Sun. 2018. Denoising Distantly Supervised Open Domain Question Answering. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1736–1745.  [10] Bernhard Kratzwald and Stefan Feuerriegel. 2018. Adaptive Document Retrieval for Deep Question Answering. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 576–587.  [11] 杨志明,时迎成,王泳,潘昊杰,毛金涛.基于BiDAF多文档重排序的阅读理解模型[J].中文信息学报,2018,32(11):117-127.  [12] Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading Wikipedia to Answer Open Domain Questions. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1870–1879.  [13] Shuohang Wang, MoYu, Xiaoxiao Guo, ZhiguoWang, Tim Klinger, Wei Zhang, Shiyu Chang, Gerald Tesauro, Bowen Zhou, and Jing Jiang. 2018. Rˆ3: Reinforced Reader-Ranker for Open-Domain Question Answering. *In Association for the Advancement of Artiﬁcial Intelligence (AAAI)*.  [14] Shuohang Wang, Mo Yu, Jing Jiang, Wei Zhang, Xiaoxiao Guo, Shiyu Chang, Zhiguo Wang, Tim Klinger, Gerald Tesauro, and Murray Campbell. 2017. Evidence Aggregation for Answer Re-Ranking in Open-Domain Question Answering. In *International Conference on Learning Representations (ICLR).*  [15] Bernhard Kratzwald, Anna Eigenmann, Stefan Feuerriegel. 2019. RankQA: Neural Question Answering with Answer Re-Ranking. In *Association for Computational Linguistics (ACL)*, pages 6076–6085*.*  [16]李童俊. 基于深度学习的开放领域自动问答系统的研究和应用[D].北京邮电大学,2019.  [17] 杨康,黄定江,高明.面向自动问答的机器阅读理解综述[J].华东师范大学学报(自然科学版),2019(05):36-52.  [18] 刘伟杰. 基于注意力机制的机器阅读理解技术研究[D].北京邮电大学,2019.  [19] Dan Hendrycks and Kevin Gimpel. 2016. Bridging nonlinearities and stochastic regularizers with gaussian error linear units. In *International Conference on Learning Representations* *(ICLR)*.  [20] Chris Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Greg Hullender. 2005. Learning to rank using gradient descent. In International Conference on Machine learning (ICML), pages 89–96.  [21] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 2383–2392.  [22] Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don’t know: Unanswerable questions for SQuAD. In *Association for Computational Linguistics (ACL)*, volume 2, pages 784–789.  [23] He, W.; Liu, K.; Lyu, Y.; Zhao, S.; Xiao, X.; Liu, Y.; Wang, Y.; Wu, H.; She, Q.; Liu, X.; et al. 2018. Dureader: a chinese machine reading comprehension dataset from real-world applications. In *Association for Computational Linguistics (ACL)*, volume 2, pages 37–46.  [24] Shao C C , Liu T , Lai Y , et al. DRCD: a Chinese Machine Reading Comprehension Dataset[J]. 2018.  [25] Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. 2013. Semantic parsing on Freebase from question-answer pairs. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1533–1544.  [26] Dhingra B , Mazaitis K , Cohen W W . Quasar: Datasets for Question Answering by Search and Reading[J]. 2017.  [27] Alexander Miller, Adam Fisch, Jesse Dodge, Amir Hossein Karimi, Antoine Bordes, and Jason Weston. 2016. Key-Value Memory Networks for Directly Reading Documents. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1400– 1409.  [28] Petr Baudis and Jan Sedivy. 2015. Modeling of the Question Answering Task in the YodaQA System. In *International Conference on Experimental IR Meets Multilinguality, Multimodality, and Interaction*, pages 222–228.  [29] Liang Pang, Yanyan Lan, Jiafeng Guo, Jun Xu, Lixin Su and Xueqi Cheng.2019. HAS-QA:Hierarchical Answer Spans Model for Open-domain Question Answering. In *AAAI Conference on Artificial Intelligence(AAAI)*.  [30] MingYan, JiangnanXia, ChenWu, BinBi, Zhongzhou Zhao, Ji Zhang, Luo Si, Rui Wang, Wei Wang, Hai qing Chen.2019. A Deep Cascade Model for Multi-Document Reading Comprehension. In *AAAI Conference on Artificial Intelligence(AAAI)*. |
| |  |  |  |  | | --- | --- | --- | --- | | 姓 名 | 职 称 | 职务 | 工 作 单 位 | |  |  |  |  | |  |  |  |  | |  |  |  |  |   **评审小组** |

|  |
| --- |
| **导师评语** |
| 导师：  日期： 年 月 日 |
| **阶段报告小组意见：** |
| 负责人：  日期： 年 月 日 |
| **学院意见：** |
| 负责人：  日期： 年 月 日 （签章） |