北京大学学报(自然科学版) 第 52 卷 第 1 期 2016 年 1 月
Acta Scientiarum Naturalium Universitatis Pekinensis, Vol. 52, No. 1 (Jan. 2016)
doi: 10.13209/j.0479-8023.2016.017

基于文本蕴含的选择类问题解答技术研究

王宝鑫 郑德权† 王晓雪 赵姗姗 赵铁军

哈尔滨工业大学计算机科学与技术学院, 哈尔滨 150001; †通信作者, E-mail: dqzheng@mtlab.hit.edu.cn

摘要 利用选择类问题具有明确候选项的特点, 简化问题分类过程, 并针对长文本语义蕴含短文本语义的语言现象, 提出一种根据文本蕴含强度大小对候选答案进行排序的方法。在没有大规模问答对的情况下, 采用维基百科中文语料库, 以全国各省市高考地理选择题作为实验数据, 通过句子相似度和文本蕴含两种方法来解答地理选择题。实验表明, 基于文本蕴含方法的准确率为36.93%, 比基于词嵌入的句子相似度方法提高2.44%, 比基于向量空间模型的句子相似度方法提高7.66%, 验证了该文本蕴含强度计算方法的有效性。

关键词 文本蕴含; 选择题; 词嵌入; 句子相似度

中图分类号 TP391

Multiple-Choice Question Answering Based on Textual Entailment

WANG Baoxin, ZHENG Dequan†, WANG Xiaoxue, ZHAO Shanshan, ZHAO Tiejun

School of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001;
† Corresponding author, E-mail: dqzheng@mtlab.hit.edu.cn

Abstract This paper proposes a method to compute textual entailment strength, taking multiple-choice questions which have clear candidate answers as research objects, aiming at the phenomenon of long text entailing short text. Two methods are used to answer the college entrance examination geography multiple-choice questions based on the Wikipedia Chinese Corpus in the absence of large-scale questions and answers. One is based on the sentence similarity and the other is based on the textual entailment proposed above. The accuracy rate of the proposed method is 36.93%, increasing by 2.44% than the way based on the word embedding sentence similarity, increasing 7.66% than the way based on the Vector Space Model sentence similarity, which confirm the effectiveness of the method based on the textual entailment.

Key words textual entailment; multiple-choice question; word embedding; sentence similarity

问答系统通常分为三类: 基于知识库的问答系统、基于大规模文本的问答系统和基于问答对的问答系统[1]。随着互联网的快速发展以及电子文本的增多, 社区问答系统(community question answering, CQA)和基于大规模文本的问答系统的相关研究不断增多, 但是针对选择题这类对人们日常生活和学习影响较大的问答系统的研究相对较少。

本文对具有明确候选项的选择题问答系统进行研究, 利用大规模维基百科中文语料作为数据源, 提出一种根据文本蕴含强度大小对候选答案进行排序的方法, 利用选择题选项的规范性来确定问题分类, 降低了问题分析过程的复杂度。最后将本文的方法与传统的句子相似度计算方法进行比较。

1 相关工作

1.1 文本蕴含相关工作

文本蕴含[2]是一个连贯文本 T 与一个假设文本 H 之间的一种关系, 如果假设文本 H 的语义可以通过文本 T 推断出来, 则认为文本 T 蕴含文本 H。文本蕴含由 Dagan 等[2]在 2004 年提出, 其相关的任

国家自然科学基金(61173073)和 863 计划(2015AA015405)资助
收稿日期: 2015–06–19; 修回日期: 2015–08–17; 网络出版日期: 2015–09–29

务一般包含识别、产生和抽取，其中关于文本蕴含识别(recognize textual entailment, RTE)的相关研究相对较多，RTE 在问答系统、信息抽取、机器翻译评测等很多应用中起关键作用[3]。RTE 常采用的方法有单独基于词汇、句法、浅层语义的无监督方法和基于分类器的有监督学习方法等[4]。有监督方法往往需要较多训练数据，并且对于训练数据的领域依赖性较强，因此本文采用基于词汇的无监督方法。以往对文本蕴含识别的研究多集中在两个句子之间，评测的任务也仅仅是评估句子 T 是否蕴含句子 H。本文文本蕴含识别则是集中在长文本与短语之间、长文本与句子之间。实际上，两个文本之间是否存在蕴含关系很难分清界限，所以现有的文本蕴含识别系统多是根据某一确定标准来判断两个句子是否存在蕴含关系。由于本文研究的是已有明确候选答案的选择题类问答系统，需要比较文本 T 对文本 H_1 的蕴含关系是否大于文本 T 对文本 H_2 的蕴含关系，而不是简单地判断两个文本之间是否存在蕴含关系。因此，为衡量蕴含关系的大小，本文提出文本蕴含强度的概念。

1.2　问答系统相关工作

问答系统一般包含 3 个主要组成部分：问题分析、信息检索和答案抽取。依据处理数据的格式，问答系统可以划分为三类：基于知识库的问答系统、基于自由文本的问答系统和基于问题答案对的问答系统。早期的问答系统大部分是基于知识库的问答系统，但是由于知识库构建需要消耗大量的资源，产生的问答系统局限性也比较大，所以该类问答系统多用来解决限定领域的问题。随着互联网的兴起，网络上的文本数量激增，随之兴起的是基于自由文本的问答系统，即从已经存在的非结构化文本中抽取答案。自 2005 年末以来，随着 CQA 数据的大量出现，问题答案对数量的增多[5]，基于问答对的问答系统逐渐成为研究热点。

本文采用全国各省市高考地理选择题作为实验数据，进行关于选择题问答系统的研究。由于知识库的匮乏，构建知识库需要消耗大量人力和时间，且关于高考题的问答对的数目相对较少，重复问题出现的可能性低，因此本文采用依赖于自由文本的问答系统。本文的选择题问答系统可以看做问答对类和自由文本类问答系统的结合：一方面，它与 CQA 一样拥有天然的候选答案可供选择；另一方面，该系统通过自由文本对选择题进行解答。传统的基于自由文本的问答系统由于没有天然可靠的候选答案，所以问题研究的重点多集中在对问题精细分类、从文本中检索相关信息以及从文本中抽取简洁的答案等方面。本文中涉及的选择题问答，由于候选选项已经确定，所以重点研究如何对候选项进行评分排序。本文采用计算文本蕴含(textual entailment, TE)强度的方法来解决选择题型问答。

2　算法与理论推导

2.1　问题定义

定义 1　文本蕴含强度。

对于一个连贯文本 T 与一个假设文本 H，如果可以根据 T 推断出 H，则说明 T 与 H 之间存在一个有向的文本蕴含关系。过去对于文本蕴含的研究多集中于两个文本 T 与 H 是否含有蕴含关系，然而在很多实际任务中，不仅需要定性地判断两个文本之间是否存在蕴含关系，而且在 T 不蕴含 H 的情况下，可能还需要判断 T 是否部分蕴含 H，以及部分蕴含多少[6]。例 1 给出一个部分蕴含的示例。

例 1　T: 李娜出生于 1982 年，是中国著名网球运动员。

H: 李娜是中国女子网球运动员。

在例 1 中可以看到，从 T 句中可以推断出 H 句的部分信息，然而并不能推断出 H 句的全部信息，其中“女子”这一信息无法从 T 句中推断出来。

针对此现象，本文提出文本蕴含强度的概念，文本 T 对 H 的文本蕴含强度指 H 与 T 之间信息的交集占 H 全部信息的比重，即连贯文本 T 对假设文本 H 的蕴含关系的大小。

定义 2　长文本蕴含。

过去针对文本蕴含的研究，多是判断两个句子之间的蕴含关系。然而实际问题中，可能会出现需要判断长文本(多个句子)对一个句子的文本蕴含关系，即长文本蕴含。例 2 给出一个长文本对单句的语义蕴含示例。

例 2　T: 李娜，1982 年 2 月 26 日出生在湖北省武汉市，中国女子网球运动员。2008 年北京奥运会女子单打第四名。

H: 网球运动员李娜在 2008 年北京奥运会获得女子单打第四名。

显然从文本 T 可以推断出文本 H，因此文本 T 蕴含文本 H。然而文本 T 包含两个句子，每个句子分别包含一部分文本 H 的信息，过去 RTE 的很多

研究方法对于该类问题并不适用。

RTE 常常采用有监督的机器学习算法，将其作为一个分类任务进行解决，但是在文本 T 是多个句子的情况下，很多特征对该类问题并不适用，并且需要人工标注较多的训练数据(长文本蕴含的标注往往需要消耗更多的时间和人力)。Glickman 等[7]采用基于词对齐的产生式模型，计算文本蕴含关系，但是他们只考虑了词之间的共现关系而忽视了词语语义、词语位置等信息。Jijkoun 等[8]利用词语相似度的方法来识别两个句子的语义蕴含关系，但其语义相似度是基于 WordNet 计算的，有一定局限性，并且也没有考虑词语位置的关系。本文改进了文献[7–8]的算法，提出一个启发式算法对文本蕴含强度进行求解。

2.2 文本蕴含强度计算方法

文本 T 对文本 H 的蕴含强度大小 TES(Textual Entailment Strength)满足式(1):

$$\mathrm{TES}(T,H)=\sum_{j=1}^{m}\lambda_j\cdot\underset{1\leqslant i\leqslant n}{\mathrm{Max}}\{\mathrm{Sim}(w_i,v_j)\},\qquad(1)$$

其中，n 表示连贯文本 T 的词数，m 表示假设文本 H 的词数，Sim 表示文本 T 中的词 w_i 对文本 H 中的词 v_j 语义蕴含的大小，λ_j 表示词语 v_j 对应蕴含强度占总蕴含强度的权重。本文用 w_i 与 v_j 之间的相似度来近似估计 w_i 对 v_j 的语义蕴含大小。

可以这样理解式(1): 对于文本 H 中的每个词 v_j，找到在文本 T 中与它相似度最高的词 w_i，计算 v_j 与 w_i 之间的相似度，最后再对所有词语相似度加权平均，求得文本蕴含强度。其中 v_j 与 w_i 的关系相当于一种词对齐关系，如图 1 所示。

λ_j 的计算过程如下: 定义 $P(H)$表示文本 H 出现的概率，$P(v_j)$表示词语 v_j 所在文本出现的概率，$P(H|v_j)$表示在词语 v_j 出现的情况下，文本 H 出现的概率。直观上，$P(H|v_j)$越大，v_j 在公式中所占的比重越大。

由贝叶斯公式(式(2))可知，当 $P(v_j|H_j)=1$，$P(H)$为定值时，$P(H|v_j)$与 $\frac{1}{P(v_j)}$ 成正比。$\log\frac{1}{P(v_j)}$ 恰好是 IDF (inverse document frequency)，常用来表示一个词语对文本的区分度。本文使用式(3)所示的归一化 IDF 作为权重λ_j。

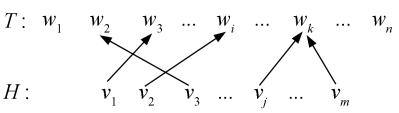

图 1 文本蕴含的词对齐表示

Fig. 1 Words alignment representation for textual entailment

$$P(H\mid v_j)=\frac{P(v_j\mid H)P(H)}{P(v_j)},\qquad(2)$$

$$\lambda_j=\frac{\mathrm{IDF}_j}{\sum_{i=1}^{m}\mathrm{IDF}_i}。\qquad(3)$$

传统词义相似度计算多是通过 WordNet 和 HowNet 等知识库计算的，因此词义相似度的计算效果往往会受限于知识库的大小。近几年，基于神经网络的 Word Embedding 因其在词语语义表示方面的良好性能受到广泛关注[9–11]。Word Embedding 将语料库中的每个词表示为一个低维实数向量，可以很好地表示两个词语语义之间的距离。Glickman 等[7]的方法需要计算任意两个词语在一句话的共现次数，往往需要较大的空间开销。Word Embedding 也利用了词共现的信息，并且能更好地表达一个词语的语义。因此，本文中的相似度是采用 Word Embedding 计算余弦相似度得到的，余弦相似度的计算如下:

$$\mathrm{Cos}(w,v)=\frac{\sum_{i=0}^{n}w_iv_i}{\sqrt{\sum_{i=0}^{n}w_i^2}\sqrt{\sum_{i=0}^{n}v_i^2}}。\qquad(4)$$

将式(1)~(4)的过程进行总结，得到算法 1。

算法1 基于词语相似度的文本蕴含强度计算。

初始化:

$T=(w_1,w_2,...,w_n)$

$H=(v_1,v_2,...,v_m)$

总相似度 totalSim= 0

总权重 totalWeight= 0

1 for j = 1, ..., m do

2 maxSim = $\underset{1\leqslant i\leqslant n}{\mathrm{Max}}\{\mathrm{Sim}(w_i,v_j)\}$

3 totalSim+ = IDF(v_j) maxSim

4 totalWeight+ = IDF(v_j)

5 end for

6 文本蕴含强度 TES= totalSim/totalWeight

7 Return TES

2.3　算法改进

算法 1 虽然可以在一定程度上表达文本蕴含关系，但是没有考虑词语位置信息。当文本 T 过长时，如果文本 H 中相邻的两个词在文本 T 中所对应的词之间的距离很大，那么 T 与 H 的词语之间的语义蕴含强度相应降低，如例 3 所示。

例 3　T: 新月与满月时，太阳、地球、月球呈一直线，潮差最大，称作大潮；上下弦月时，三者呈直角，潮差最小，称为小潮。

H_1: 地球处在太阳与月球之间，出现大潮。

H_2: 地球处在太阳与月球之间，出现小潮。

对于例 3，显然文本 T 对 H_1 的文本蕴含强度应该大于 T 对 H_2 的蕴含强度。事实上，从文本 T 可以推断出 H_1，而无法推断出 H_2。因此，我们提出对应的改进算法，相应的蕴含强度计算如下:

$$\mathrm{TES}(T,H)=\sum_{j=1}^{m}\lambda_j \max_{1\leqslant i\leqslant n,1\leqslant j\leqslant m}\{F(p_1(w_i)-p_2(v_{j-1}))\mathrm{Sim}(w_i,v_j)\}, \tag{5}$$

其中，m 和 n 分别表示假设文本 H 和连贯文本 T 的词数，$p_1(w_i)$ 表示词语 w_i 在文本 T 中所在的位置下标，$p_2(v_{j-1})$ 表示词语 v_{j-1} 在文本 T 中对应词所在的位置下标，即 $p_1(w_i)-p_2(v_{j-1})$ 是文本 T 中的两个词之间的距离。

文本 H 中相邻的两个词所对应的文本 T 中的两个词距离越远，其语义蕴含强度越低，且这种降低趋势随距离增大先缓慢降低，到一定距离后再加速降低，最后再缓慢降低，高斯函数 $F(x)$ (式(6))正好满足这种下降趋势。

$$F(x)=\exp\left(\frac{-(x-b)^2}{2\sigma^2}\right), \tag{6}$$

其中，x 是变量，对应两个词 w_i 和 w_k 之间的距离 $p_1(w_i)-p_2(v_{j-1})$；b 和 σ 是高斯函数 $F(x)$ 的参数，b 的大小取决于词 w_i 和 w_k 的位置关系，显然 w_i 和 w_k 相邻时，$F(x)$ 最大，即它们之间的最佳距离取 1 比较合适。本文中 b 取值为 1。

我们用动态规划求解获得最终 TES 的值，具体描述如算法 2 所示。

算法 2　改进的文本蕴含强度计算

输入: 连贯文本 $T=(w_1,w_2,...,w_n)$

假设文本 $H=(v_1,v_2,...,v_m)$

输出: 文本蕴含强度 TES

1 初始化:

$\delta(i,0)=0,\quad i=1, 2, ..., n$

2 递推:

$$\delta(i,j)=\max_{1\leqslant k\leqslant n}\{\delta(k,j-1)+\lambda_j F(i-k)\mathrm{Sim}(w_i,v_j)\}$$

3 终止:

Return $\max\{\delta(i,m)\}\quad i=1, 2, ..., n$

3　选择类问题解答及分析

鉴于高考地理题具有易获取、少干扰、形式规范以及可靠性高的特点，本文采用各地高考近十年的地理选择题，去除其中含有图片的题目以及计算类题目，剩余 287 道选择题作为最终的实验数据。

本文方法分为预处理、问题分析、信息检索与答案抽取 4 个模块，如图 2 所示。

3.1　预处理

预处理阶段，对维基百科文本语料进行分词，并用分词后的维基百科中文文本语料和 Mikolov 等[10–11]提出的 word2vec 工具实现 Word Embedding 的训练。使用目前国际上句法分析效果比较好的 ZPar[12]工具，对选择题选项进行句法分析。

3.2　问题分析

3.2.1　关键词抽取

本文通过传统的 TF-IDF 方法来提取关键词，即根据计算选择题题干部分的 TF-IDF 的数值大小进行排序，去除停用词后，依据 TF-IDF 值的大小依次选取关键词，本文实验中选取的关键词数目为 3。例 4 是一道高考地理选择题的实例。例 5 是针对例 4 的一个抽取关键词的例子。从例 5 可以看出，基于 TF-IDF 抽取关键词的方法虽然简单，但是在地理选择题题干中的表现很好。

例 4　春季，欧洲阿尔卑斯山区，背风坡常常出现冰雪迅速融化或雪崩。其主要原因是

A. 反气旋控制下沉增温

B. 暖锋过境释放热量

C. 西风带南移释放热量

D. 局地气流下沉增温

例 5　题干: “春季，欧洲阿尔卑斯山区，背风坡常常出现冰雪迅速融化或雪崩。其主要原因是”。抽取关键词: 背风坡、阿尔卑斯、雪崩。

3.2.2　问题分类

传统问答系统的问题分类通常比较精细，一方面为了确定答案的类型，同时也为了对不同类别的

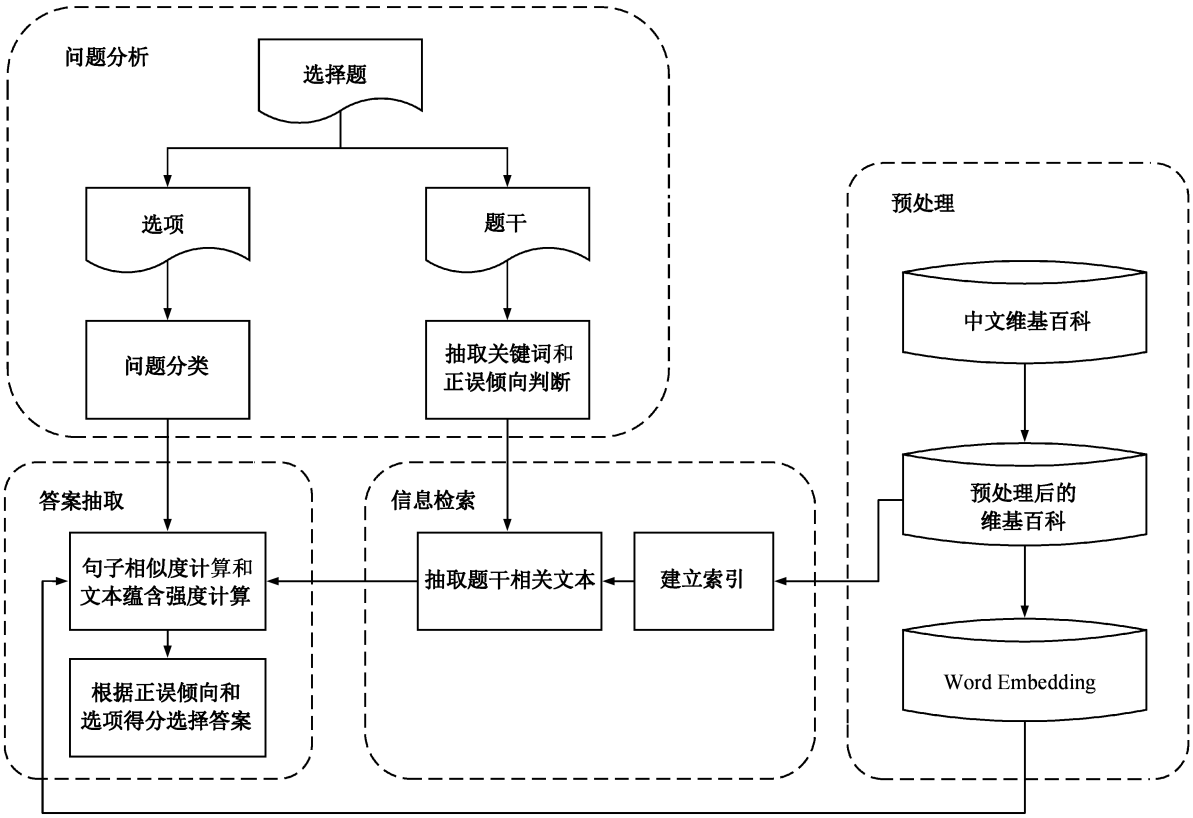

图 2 选择题问答系统框架

Fig. 2 Frame diagram of multiple-choice question and answering

问题采用不同的方法来解答。本文采用的高考题具有规范性，候选答案的形式规范且符合问题要求。根据该特点，依据选择题的选项对问题分为两大类：一类是候选答案为名词短语的选择题；另一类是候选答案为句子的选择题。本文对选项的分析判断采用句法分析，4 个选项中含有名词短语(NP)的选项有两个及两个以上则为名词短语类型，否则即为句子类型(IP)。

例 6 是一道地理选择题，其中的 4 个选项都是 NP，因此该选择题将会被划分为名词短语类型。

例 6 人类已知月球上的能源有

A. (NP (NN 生物能) (PU 、) (NN 风能))

B. (NP (NN 核能) (PU 、) (NN 潮汐能))

C. (NP (NN 潮汐能) (PU 、) (NN 太阳能))

D. (NP (NN 太阳能) (PU 、) (NN 核能))

3.2.3 问句正误倾向分析

选择题经常会要求判断“不正确”、“错误”或“不合理”。对于这类问题，我们将其识别出来，为后面的答案抽取过程提供帮助。该部分主要通过人工配置词典的方法，对选择题题干进行识别，例如，在题目的问句中出现“不正确”一词，则将该问题作为错误倾向类的问题。

3.3 信息检索

对中文维基百科的词条建立索引，根据问题分析阶段抽取出来的关键词，在维基百科语料中检索相应的词条，将与其对应的百科文本提取出来。

3.4 答案抽取

该阶段分别采用句子相似度和文本蕴含两种方法来实现答案抽取。最后根据问题分析中的正误倾向性判断来选择答案。如果是正向问题，则选择分值最高的选项，否则，选择分值最低的选项。

3.4.1 句子相似度

在中文维基百科文本中检索关键词对应的百科文本，将选项与百科文本中的所有句子一一进行相似度计算，选取最高的相似度作为该选项最终的分数。相似度计算分别采用基于 TF-IDF 的向量空间模型和基于 Word Embedding 的句子相似度计算。

基于VSM的句子相似度：将两个句子表示为两

个向量，向量的每一维权值对应每个词的 TF-IDF 值，再对两个向量计算余弦相似度，作为两个句子最终的相似度。

基于 Word Embedding 的句子相似度：如式(7)和(8)所示，将句子中每个词的 Word Embedding 向量 w_i 相加取平均值作为句子的向量 $\boldsymbol{s}$，再对两个句子的向量计算余弦相似度，作为两个句子最终的相似度。

$$\boldsymbol{s}=\frac{1}{n}\sum_{i=1}^{n}\boldsymbol{w}_i\ , \tag{7}$$

$$\mathrm{Sim}(T,H)=\cos(\boldsymbol{s}_T,\boldsymbol{s}_H)\ 。 \tag{8}$$

3.4.2　文本蕴含

将关键词对应的维基百科文本整体作为文本 T，句子选项作为文本 H，对短语类的问题采用算法 1，对句子类的问题采用算法 2，计算 T 对 H 的文本蕴含强度。

4　实验结果与分析

由于本文问答系统中候选项已经确定，正确答案一定会出现在候选项中，且每道题都有固定的 4 个候选项，所以本文对问答系统的评测标准采用准确率。算法 2 中高斯函数 $F(x)$ 的参数设置如下：$b=1$，$\sigma=5$。

根据句子相似度和文本蕴含得到的最终问答系统准确率如表 1 所示。从表 1 可见，基于 Word Embedding 的相似度计算方法好于基于 VSM 的方法。可见基于 Word Embedding 的方法比 VSM 的方法能更好地表达句子的语义。从表 1 还可以看出，算法1对名词短语类的问题效果比较好，而算法 2 对于句子类的问题效果较好。综合两种方法后，本文提出的方法最终的准确率可达 36.93%。

为了验证本文方法的有效性，在选取关键词对应的百科全部文本作为连贯文本 T 之外，还将百科文本中不同数目的连续句子作为 T 进行实验，选取其中最大的文本蕴含强度作为最终选项的分值。

表 1　实验结果

Table 1　Experiment result

实验方法	名词短语类问题(NP)/%	句子类问题(IP)/%	全部问题(ALL)/%
VSM	30.71	27.89	29.27
Word Embedding	36.43	32.65	34.49
算法 1	38.57	27.21	32.75
算法 2	32.14	35.37	33.80
综合算法 1 和 2	38.57	35.37	36.93

图 3 是对应的实验结果，可以看出，算法 1 对应名词短语类问题的解答准确率随着句子数目的增大而呈上升趋势，但是算法 1 却无法对候选项为句子的问题进行有效解答。随着句子数目增大，算法 1 对句子类问题逐渐失效。原因可能有以下两点：1)算法 1 无法很好地分析含有完整句法结构的句子所对应的文本蕴含情况；2)词短语部分的选择题更倾向于概念类题目，相对简单，而候选答案为句子的选择题分析则较为复杂，需要更深层的语义分析，因此无法直接从百科抽取答案。

例 7 是在算法 2 中正确而在算法 1 中错误的一个例子(算法 2 的答案为 D，算法 1 的答案为 A)，其对应的候选项都为句子。例 7 在一定程度上反映了算法 2 对候选项为句子的问题的解答效果比算法 1 好。

例 7　在森林中一旦遭遇火灾，下列做法正确的是

A. 使用沾湿的毛巾遮住口鼻，顺风逃离

B. 如果火势突然减弱，则可以放心休息

C. 选择低洼地或坑洞躲避

D. 伺机逆风突破林火包围

算法 2 在名词短语类问题上的表现不如算法 1，原因可能是名词短语类选项大多由多个实体名词混合在一起组成，在百科文本中出现的位置相对分散，限制其位置会导致最终的准确率较低。算法 2 对于候选项为句子的问题解答效果显然比算法1 好很多，并且其准确率随着句子数目增多而增大，这也说明

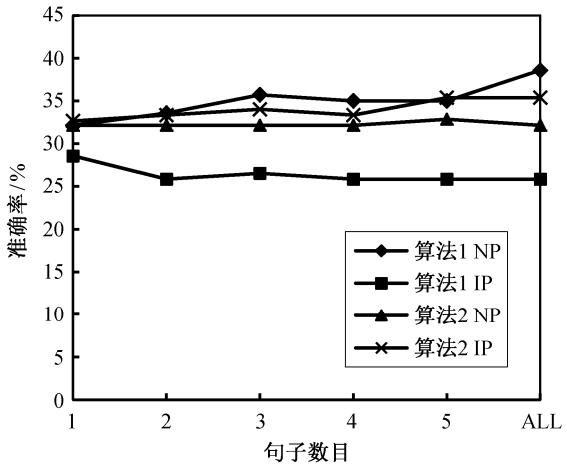

图 3　不同句子数目的准确率

Fig. 3　Accuracy of a different number of sentences

算法 2 对于计算长文本对句子的文本蕴含强度的效果明显。

5 结论

本文针对选择类问题解答方法进行了研究，提出了一种新的计算文本蕴含强度的方法。在没有大规模训练数据的情况下，仅用维基百科中文语料库，通过 Word Embedding 计算文本蕴含强度来解决地理选择类问题，最终基于文本蕴含方法的准确率为 36.93%，比基于 VSM 的句子相似度方法的准确率高 7.66%，比基于 Word Embedding 的句子相似度方法高 2.44%。实验验证了本文提出的文本蕴含计算方法对长文本蕴含短文本的情况效果明显，并且文本蕴含也是解答选择类问题的有效的方法。

由于本文关于文本蕴含强度的计算方法是分别针对长文本对短语和长文本对句子两种类型的文本蕴含情况进行的，所以该方法在句子对句子类型的文本蕴含强度的计算效果仍有待提升。此外，对于推理类地理选择题，本文的方法在很多情况下并不适用，需要后期构建大型的知识库以及逻辑推理框架来解决。

参考文献

[1] 毛先领，李晓明. 问答系统研究综述. 计算机科学与探索, 2012, 6(3): 193–207

[2] Dagan I, Glickman O. Probabilistic textual entailment: generic applied modeling of language variability // Proc of the Pascal Workshop on Learning Methods for Text Understanding & Mining. Grenoble, 2004: 26–29

[3] Androutsopoulos I, Malakasiotis P. A survey of paraphrasing and textual entailment methods. Journal of Artificial Intelligence Research, 2009, 38(4): 135–187

[4] 袁毓林，王明华. 文本蕴涵的推理模型与识别模型. 中文信息学报, 2010, 24(2): 3–13

[5] 张中峰，李秋丹. 社区问答系统研究综述. 计算机科学, 2010, 37(11): 19–23

[6] Levy O, Zesch T, Dagan I, et al. Recognizing partial textual entailment // Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics. Sofia, 2013: 451–455

[7] Glickman O, Dagan I M. A lexical alignment model for probabilistic textual entailment // Machine Learning Callenges: Evaluating Predictive Uncertainty, Visual Object Classification, and Recognising Tectual Entailment. Berlin: Springer, 2006: 287–298

[8] Jijkoun V, de Rijke M. Recognizing textual entailment using lexical similarity // Proc of the First PASCAL Challenges Workshop on RTE. Southampton, 2005: 73–76

[9] Collobert R, Weston J. A unified architecture for natural language processing: deep neural networks with multitask learning // Proceedings of the 25th International Conference on Machine Learning. Helsinki, 2008: 160–167

[10] Mikolov T, Chen K, Corrado G, et al. Efficient estimation of word representations in vector space // Proceedings of the Workshop at ICLR. Scottsdale, 2013: 1–12

[11] Mikolov T, Sutskever I, Chen K, et al. Distributed representations of words and phrases and their compositionality // Proceedings of Neural Information Processing Systems. Lake Tahoe, 2013: 3111–3119

[12] Zhang Y, Clark S. Syntactic processing using the generalized perceptron and beam search. Computational Linguistics, 2011, 37(1): 105–151