文章编号：2095-2163(2019)01-0192-07 中图分类号：TP301 文献标志码：A

中文复述问句生成技术研究

曹 雨，张 宇，刘 挺
（哈尔滨工业大学 计算机科学与技术学院，哈尔滨 150001）

摘 要： 自动问答系统允许用户以自然语言进行提问，问题的形式多样、结构复杂，对系统的理解能力提出了极高要求。问句复述生成技术可将提出的复杂问句改写成一系列与之语义相同但形式不同的问句，避免了用户提问的不规范，可大大降低系统对问句的理解和处理难度，对于提升自动问答系统的效果有着重要意义。本文提出了一种基于模板匹配的复述问句生成方法，该方法可有效保留问句的结构特征和语义特征。引入功能标签，突出问句的结构特征；引入依存关系，提高了问句模板的泛化性能；引入候选排序，大幅提升了生成结果的准确率。通过与已有的生成方法进行对比试验，证实了该方法的有效性。

关键词： 问答；复述生成；模板匹配；候选排序

Research on Chinese question paraphrase generation techniques for question answering

CAO Yu, ZHANG Yu, LIU Ting
(School of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001, China)

【Abstract】 The question answering system allows the user to ask questions in natural language, the form of the query is diverse and the structure is complex, the system's understanding ability is challenged. The paraphrase generation technique can be used to paraphrase the complicated question into a series of different questions with the same semantics, which could avoid the non-standard of the user's question and reduce the difficulty of understanding and processing, and improve the performance of QA system. This paper proposes a method of question paraphrase generation based on template matching. By using the method, the structural and semantic features of the question could be effectively preserved. The functional labels can highlight the structural features; the dependency can improve the generalization of the question template; the candidate ranking can greatly enhance the precision of the generation. The experimental results prove the efficiency of this method.

【Key words】 QA; paraphrase generation; template matching; candidate ranking

0 引 言

复述生成技术在很多方面有着重要应用，如：在问答系统中，可用来提升系统对问句的理解能力，优化系统性能；在机器翻译领域[1]，可用来扩展语料规模，解决数据稀疏问题；在阅读理解任务中，可用于问句多样性的研究，使机器自动生成的问题更符合人类的自然语言规范。

复述模板的表示和抽取[2]是研究复述的重要方法之一，合适的复述模板表示方法和复述模板自动抽取方法是主要难点。复述模板拥有很强的表达能力，能够有效地表达自然语句的结构和特征，可以用来进行大量复述实例的生成。目前已存在很多种复述模板的表达方法，如：利用词的词性作为特征表示复述模板；将某些词语替换成变量作为复述模板。本研究从句法分析入手，结合词性、命名实体和功能词等信息，形成了一种新的复述模板抽取的方法，同时也保留了句子的结构信息、语义信息以及每个词对应的上下文信息。在该方法中，每条中文问句对应一个句子模板，从而每组复述语料可对应一组模板，再结合匹配生成方法，生成一系列的候选生成句，最后利用基于相似度计算和语言模型相结合的方式，对候选生成句进行打分排序。

本文引入现有较好的其它基于问句模板的复述生成方法以及基于深度学习模型的复述生成方法作为对比，进行实验。实验结果证明了本文所提出的方法的有效性，可以大大提升复述生成的准确率。

1 相关工作

近年来，学界已陆续涌现了许多研究方法与模

作者简介：曹 雨(1992-)，男，硕士研究生，主要研究方向：自然语言处理、问句复述生成；张 宇(1979-)，男，博士，副教授，硕士生导师，主要研究方向：网络安全、网络测量、未来网络等；刘 挺(1972-)，男，博士，教授，博士生导师，主要研究方向：自然语言处理、文本挖掘、文本检索等。

收稿日期：2017-06-12

型，并且均取得了可观成果，但这些模型与方法也都有着各自的弊端，有些准确率偏低，如：Lin 等人[3]提出的 DIRT 方法；有些规模受限、领域受限，如：基于机器翻译评测语料的复述句抽取方法；有些生成的结果偏于复杂，如：Pang 等人[4]提出的有穷自动机方法；有些复述的来源受到限制，如：Zhou 等人[5]提出的翻译特征方法。

目前，虽然国内对于复述的研究日渐重视，但与国外相比中文的复述研究仍亟待完善，尤其是针对开放领域问答系统的问句复述技术的研究上，面临着较大的研究挑战。仅就问句分析技术而言，因其作为自动问答系统中至关重要的组成部分，分析效果的好坏将对整个问答系统的表现发挥决定性的影响作用。其中，针对问句分析模型可以理解和分析更多的可变长度的问句的问题，在很长一段时间内，主要遵循了 2 种途径方法：问句扩展和复述生成；基于句法结构复述生成方法，与之前 2 种方法相比可以生成更多可扩展的问句，但前述方法多数情况下会用于英文语料的研究，究其原因则在于中文语料存在以下 3 个难点：

(1)中文语料不足，且规模较小。

(2)不存在面向中文语料的语法分析器。

(3)中文语法过于灵活和复杂。

机器学习和深度学习技术的迅速发展，也为复述生成技术的研究提供了更多可能的有效解决方法。但其对语料的要求较高，需要大规模高精度的中文复述资源库，针对开放领域问答系统的研究，需要大量的高精度单语复述问句句对，因此中文问句复述语料规模的限制成为问句复述技术研究应用于开放领域问答系统的瓶颈。

2　基于模板匹配的复述生成方法

研究提出的方法主要分为问句模板抽取、模板匹配生成以及候选生成句排序三个模块。对于某一问句 q_{input}，首先采用一系列的预处理操作，然后通过问句模板抽取模块，抽取模板 t_{input}，再利用匹配生成模块生成候选模板集 T 和候选问句集 Q，最后利用候选生成句排序模块对候选生成问句进行排序。下面，将对每部分研究展开详述如下。

2.1　复述语料获取及预处理

本次研究主要针对开放领域，且复述语料资源有限，因此选取“百度知道”相似问题作为语料来源，共爬取百度知道完整问句 2 232 051 条、373 704 组。研究采取的具体形式可见表 1。

表 1　语料格式

Tab. 1　Format of the corpus

语料格式内容
诺基亚 5800、5230、5530 那个好？为什么
诺基亚 5800 和 5230 和 5530 哪个好??
诺基亚 5230,5800,5530 哪个好？
诺基亚 5800 5230 5530 5235 哪个好
诺基亚的 5800、5530 和 5230 哪部好用
诺基亚 5230 和 5530.5800 那个好。

语料中的很多问句较长，其中含有引用成分，如：“白日依山尽”这首诗的作者是谁。引号中的成分在模板抽取的过程中应作为一个成分，在后续的分词过程中不应进行处理。所以，研究将语料中书名号和引号中的内容进行标记，不参与分词，在模板抽取过程中直接对应<n>标签。

从线上问答系统中获取的语料资源来自于用户提出，问句中符号、标点的使用极其不规范，必将影响后续的实验研究过程，因而需要删除。但实验过程中发现，标点、符号的使用会牵涉到句子的分词、词性标注及依存关系分析，如图 1 所示，基于此，研究中将采用 LTP 平台对语料进行依存关系分析和词性标注后，再将问句资源中的符号进行删除处理。

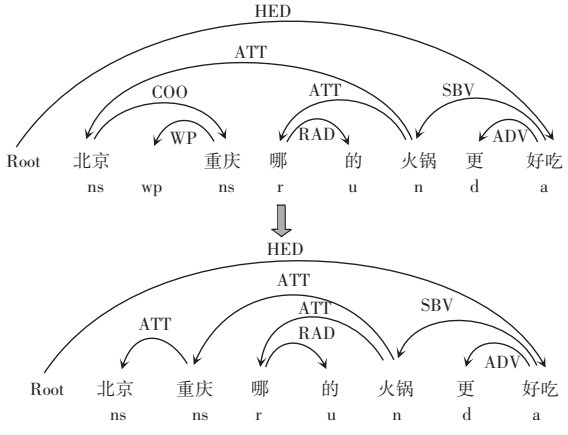

图 1　符号对依存关系分析的影响

Fig. 1　The impact of symbols on dependency analysis

语料中还有很多问句的成分是不完整的。这类问句在后续实验的问句简化过程中，句子长度会急剧减少。如：“河北城乡建设小学宿舍”，这个句子中，“河北城乡建设小学”作为“宿舍”的修饰限定，在句子成分简化过程中会被暂时移除，保留其中心词“宿舍”，无法构成一个完整的句子，对后续问句模板提取没有任何贡献。又因为构成一个完整的中文问句至少需要包括 3 个词，如：“马云是谁”，且问句长度过长难以进行有效分析。所以，对语料进行

依存关系分析,去掉其中修饰限制成分后,问句长度最短为 3,最长设定为 20。

经上述预处理过程后,研究中共保留复述语料 1 178 126 条、272 174 组。

2.2 问句模板抽取

问句模板提取主要包括 5 个部分:分词、词性标注、命名实体识别、功能标签替换以及句法分析。对此可做阐释论述如下。

采用 LTP 平台对语料进行分词、词性标注、命名实体识别,在保持句子中词汇顺序不变的条件下,用词性标签和实体标签代替其所对应的词汇,初步形成问句模板,过程解析详见表 2。

表 2 初步形成问句模板

Tab. 2 Initial form of query template

原 句	北京哪能找到好吃的西餐						
分 词	北京	哪	能	找到	好吃	的	西餐
词性标注	ns	r	v	v	a	u	n
命名实体	ns	-	-	-	-	-	-
初步模板	<ns> <r> <v> <v> <a> <u> <n>						

此时的问句模板已经具有一定的表达和泛化能力,可代表与此句式结构相同的一部分自然语言问句。但在诸如表 2 所示的实验样例中,研究观察初步生成的问句模板可以发现,“哪”表示地点疑问词,在上述过程中仅仅被简单地标注为‘r’(代词)标签,无法表征出其语义特征;“能”、“找到”两个词都被标注为‘v’标签,但这并不准确,实际上“能”属于情态动词,并不表示实际或具体的动作。因此,上述步骤是不完善的,无法真实有效地表示出某些词汇的语义特征、语法结构及句法结构。至此,为了将诸如上例所述的这部分特殊词与其它与之具有相同词性的词汇进行区分,研究专门收集了一些与表 2 实例中“哪”类似的可表示问句特征的关键词以及一些与“能”类似的情态动词,形成词表,进行人工标注,为每一个词赋予一个标签,最终形成的功能词表可见表 3。更新后的问句模板为:<ns> <where> <modal> <v> <a> <u> <n>。

中文语法灵活多变,导致某些问句结构成分复杂,难以分析。此类问句在语料中占有较大比重,经分析发现,句中含有大量的修饰限制成分,如:“中国北京有多少家好吃的餐馆”,在此例中“中国”是用来修饰“北京”的,“好吃”和“的”是用来修饰“餐馆”的,此类词汇的有无对句义不会造成很大的影响,却会在后续的模板匹配和生成过程中产生重大影响,使得同类型的句子因为修饰限制词的影响而无法成功匹配生成。综上可知,虽然无法具体去确定每一个词的修饰限制用法,但却可以就某一类词进行研究,从而找出其通常情况下用法,将句中的修饰限定成分暂时移除,简化句子结构,从而使得句子模板更具泛化性,寻找到更多匹配模板,生成更多的候选问句。

表 3 功能词表

Tab. 3 Functional vocabulary list

词 汇	标签
哪/哪儿/哪里	<where>
哪个/哪些	<which>
谁	<who>
什么	<what>
为什么	<why>
能/能够/可能/可以/…	<modal>
……	……
……	……

句子结构的简化过程极其繁琐、费时。在此过程中,并不能简单地将某一类词直接从句子中移除,如:形容词(词性标注为 a)是最常见的修饰词,通常情况下用来修饰名词(词性标注为 n),可以被移除,但通过实验及对语料的分析,发现类似于“马尔代夫和毛里求斯,哪个风景更漂亮”这样的问句,本句中“漂亮”是一个形容词,但却不能直接移除,若直接移除,则该句成分将会出现严重缺失。与此类似的句子在语料中还存在很多,如:“视频转换大师与格式工厂哪个更快”。中文句法过于灵活,其中还存在着多种修饰限制关系,如:“文化教育”为名词修饰名词;“调查研究”为动词修饰动词;“漂亮美丽”为形容词修饰形容词等等。本次研究中采用句法分析,并根据句法分析的结果构建句法树,先引入 6 种较为常见的修饰关系对语料进行简化,分析句子简化的结果,根据某种修饰关系对语料简化的影响程度,排除已有的修饰关系或引入新的修饰关系,如此反复,最后筛选确认 8 种对句子简化最为有效、且移除后不会对句子主干造成影响的修饰关系,对此描述可参见表 4。

句子成分精简后,如表 2 所示的样例的最终模板变为<ns> <where> <modal> <v> <n>。按照上述方法对语料进行句式简化和模板抽取,所得结果可见表 5。

2.3 模板匹配生成

上述模板抽取过程结束后,会生成 2 部分资源。一部分为原始语料进行句子精简后的自然问句复述

资源，另一部分为其对应的复述模板资源。将这 2 部分资源作为匹配生成过程中待查询的资源库。

输入某一新问句，作为待改写生成的问句，采用上述的句式精简和模板抽取对该问句进行处理。在对此问句的精简过程中，仅仅是暂时移除句子中的修饰限制成分，将其保留起来，待后续使用，如图 2 所示。抽取待改写问句模板，将该模板在整个复述模板库中进行全匹配检索，在某一复述组中检索到该匹配项，则证明该组其它句子具有改写成该句的可能性，将这若干组模板复述资源重组，形成新的候选复述模板资源，候选复述模板资源所对应的精简自然问句构成新的候选自然句复述资源，即如图 3 所示。

表 4　可简化修饰关系

Tab. 4　The relationship which can be simplified

可简化修饰关系
a→n(形容词修饰名词)
d→a(副词修饰形容词)
a→a && a ≠ head(形容词修饰形容词)
u→a(助词修饰形容词)
u→d(助词修饰副词)
n→n(名词修饰名词)
b→n(名词修饰语修饰名词)
j→n(缩写词修饰名词)

表 5　部分语料句式精简、模板抽取结果

Tab. 5　The template of some querys which are simplified

部分语料句式	句式精简、模板抽取结果
诺基亚 5800 5230 5530 那个 好 为什么	<nz> <m> <m> <m> <r> <a> <why>
诺基亚 5800 和 5230 和 5530 哪个 好 用	<nz> <m> <c> <m> <c> <m> <which> <a> <v>
诺基亚 5230 5800 5530 哪个 好	<nz> <m> <m> <m> <which> <a>
诺基亚 5800523055305235 哪个 好	<nz> <m> <which> <a>
诺基亚 的 5800 5530 和 5230 哪 部 好 用	<nz> <u> <m> <m> <c> <m> <where> <q> <a> <v>
诺基亚 5230 和 5530.5800 那个 好	<nz> <m> <c> <m> <r> <a>

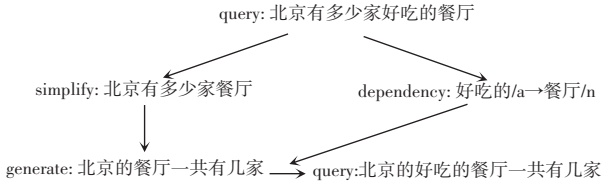

图 2　句式精简及还原过程

Fig. 2　The query simplified and recovered process

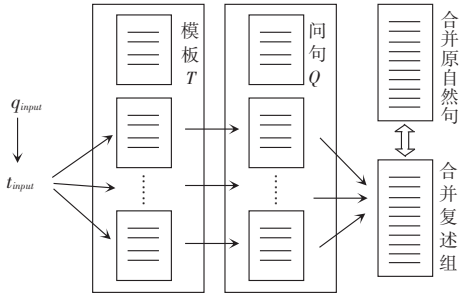

图 3　复述组合并

Fig. 3　Combining the group of paraphrases

选取候选复述模板资源中的某一条，与待改写问句模板进行比对，将其与原模板相同的标签部分作为槽，其余部分为其特征部分，保持不变，如此可形成待生成的特征模板。将待生成的特征模板与其自然问句对应，保留其特征部分的词，其余部分作为槽。将带改写问句中的词根据槽所对应的标签，依次填写进槽内，生成新的问句。其设计过程如图 4 所示。

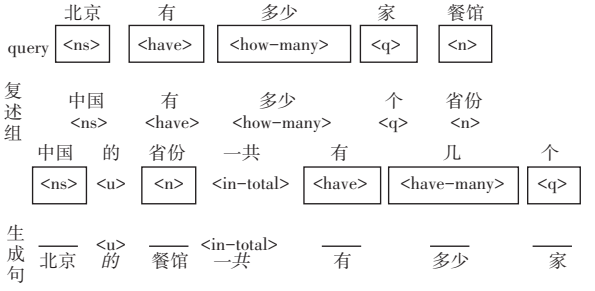

图 4　复述改写生成过程

Fig. 4　The process of paraphrase generation

2.4　候选生成句排序

本部分研究提出一种基于相似度计算与语言模型相结合的方式进行候选生成句的打分排序，其中语言模型采用 RNN-LM，相似度计算采用 Wang 等人[6]研发的基于相似与相异信息的 CNN 模型。

RNN-LM[7]利用神经网络对语言进行建模，与传统的 N-gram 相比，尤为突出的一个优点就是将历史信息映射到了一个低维的空间，从而降低了模型的参数，并将相似的历史信息进行了聚类。先用 RNN-LM 对候选生成句进行打分，记为 S_{lm}，然后对

其进行归一化操作,所得分值作为语言模型对于候选生成句的最终打分,记为 S_1。

基于相似与相异信息的相似度计算方法,可得模型设计结构如图 5 所示。研究中,首先使用由 Mikolov 等人[8]提出的模型训练出来的词向量进行句子的表示。对于句子 $\boldsymbol{S}$ 和 $\boldsymbol{T}$,则将其表示为向量矩阵 $\boldsymbol{S}=[S_0,\cdots,S_i,\cdots,S_m]$ 和 $\boldsymbol{T}=[T_0,\cdots,T_j,\cdots,T_n]$,其中 $\boldsymbol{S}$ 和 $\boldsymbol{T}$ 是句子中词汇的 d 维词向量,m 和 n 则是句子中包含词汇的数量。

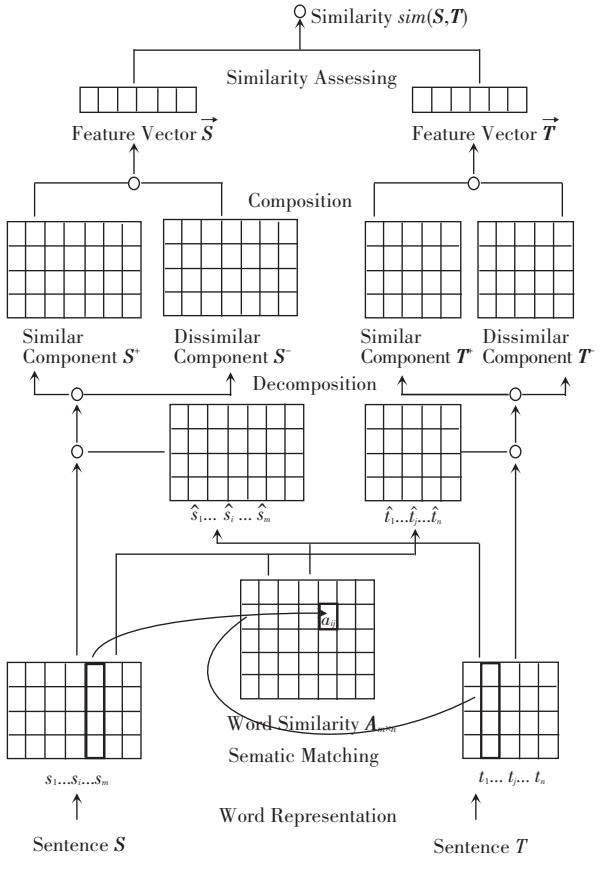

图 5 基于相似与相异信息的 CNN 模型

Fig. 5 CNN model based on similar and different information

为了计算句子 $\boldsymbol{S}$ 和 $\boldsymbol{T}$ 的相似度,即需判断句子 $\boldsymbol{S}$ 中的词能否被句子 $\boldsymbol{T}$ 中的词或短语的语义覆盖,因此通过组合 $\boldsymbol{T}$ 中的部分词向量来表示句子 $\boldsymbol{S}$ 中的每一个词 $\boldsymbol{S}_i$,得到 $\boldsymbol{S}_i$ 的语义匹配向量 $\hat{\boldsymbol{S}}_i$。

首先使用余弦相似度计算句子 $\boldsymbol{S}$ 和 $\boldsymbol{T}$ 的相似度矩阵 $\boldsymbol{A}_{m\times n}$,对于相似矩阵 $\boldsymbol{A}_{m\times n}$,矩阵中的元素 a_{ij} 是 $\boldsymbol{S}_i$ 和 $\boldsymbol{T}_j$ 的余弦相似度,则可通过式(1)进行计算:

$$a_{ij}=\cos(K_s^i,K_t^j)$$

$$\cos(X,Y)=\frac{\boldsymbol{S}_i\,\hat{\boldsymbol{S}}_i}{\|\boldsymbol{S}_i\|\cdot\|\hat{\boldsymbol{S}}_i\|} \tag{1}$$

通过 $\boldsymbol{A}_{m\times n}$ 运算得到了句子 $\boldsymbol{T}$ 中同 $\boldsymbol{S}_i$ 最相似的词汇 $\boldsymbol{T}_k$,并使用 $\boldsymbol{T}_k$ 及其上下文来表示 $\boldsymbol{S}_i$,研究推得数学公式如下:

$$\hat{\boldsymbol{S}}_i=f_{match}(\boldsymbol{S}_i,\boldsymbol{T})=\frac{\sum_{j=k-w}^{k+w}a_{ij}\,t_j}{\sum_{j=k-w}^{k+w}a_{ij}} \tag{2}$$

其中,$k=argmax_j\,a_{ij}$ 为 $\boldsymbol{T}$ 中同 $\boldsymbol{S}_i$ 最相似的词的下标。

研究中使用 $\boldsymbol{T}_k$ 及其窗口大小为 w 的上下文的词向量的加权平均来表示 $\boldsymbol{S}_i$,每个词的权值大小为该词同 $\boldsymbol{S}_i$ 的相似度,以 $\boldsymbol{T}_k$ 为例,其权值为相似矩阵 $\boldsymbol{A}_{m\times n}$ 的元素 a_{ik}。

对于 $\boldsymbol{S}_i$ 得到了其语义匹配向量 $\hat{\boldsymbol{S}}_i$,$\hat{\boldsymbol{S}}_i$ 可以看作是句子 $\boldsymbol{T}$ 对 $\boldsymbol{S}_i$ 的语义覆盖。很明显,$\hat{\boldsymbol{S}}_i$ 和 $\boldsymbol{S}_i$ 在语义上一定不同,因此利用 $\hat{\boldsymbol{S}}_i$ 将 $\boldsymbol{S}_i$ 分解为 2 个向量。一个为 $\boldsymbol{S}_i$ 与 $\boldsymbol{T}$ 的相似向量,另一个为 $\boldsymbol{S}_i$ 与 $\boldsymbol{T}$ 的相异向量。此后,研究中将使用余弦相似度计算 $\boldsymbol{S}_i$ 和 $\hat{\boldsymbol{S}}_i$ 的相似度,再基于相似度对 $\boldsymbol{S}_i$ 进行分解,推得其数学表述如式(3)所示:

$$\begin{aligned}\alpha&=Pearson(\boldsymbol{S}_i,\hat{\boldsymbol{S}}_i)\\ \boldsymbol{S}_i^+&=\alpha\,\boldsymbol{S}_i\\ \boldsymbol{S}_i^-&=(1-\alpha)\,\boldsymbol{S}_i\end{aligned} \tag{3}$$

可以看出,若 $\boldsymbol{S}_i$ 与 $\hat{\boldsymbol{S}}_i$ 越相似,则越多的部分会被分到 $\boldsymbol{S}_i^+$ 中。对 $\boldsymbol{S}$ 中的每一个词进行上述操作,将 $\boldsymbol{S}$ 分解为相似矩阵 $\boldsymbol{S}^+$ 及相异矩阵 $\boldsymbol{S}^-$:$\boldsymbol{S}^+=[\boldsymbol{S}_0^+,\cdots,\boldsymbol{S}_m^+]$,$\boldsymbol{T}^+=[\boldsymbol{T}_0^+,\cdots,\boldsymbol{T}_n^+]$。同理,也能将 $\boldsymbol{T}$ 分解为相似矩阵 $\boldsymbol{T}^+$ 与相异矩阵 $\boldsymbol{T}^-$:$\boldsymbol{S}^-=[\boldsymbol{S}_0^-,\cdots,\boldsymbol{S}_m^-]$,$\boldsymbol{T}^-=[\boldsymbol{T}_0^-,\cdots,\boldsymbol{T}_n^-]$。

这里,将使用以上信息对 $\boldsymbol{S}$ 和 $\boldsymbol{T}$ 进行建模并计算两者的相似度。受到 Kim 等人的启发,研究使用双通道的 CNN 模型对相似矩阵 $\boldsymbol{S}^+$ 及相异矩阵 $\boldsymbol{S}^-$ 进行建模,得到句子 $\boldsymbol{S}$ 的特征向量 $\boldsymbol{F}_s$ 和句子 $\boldsymbol{T}$ 的特征向量 $\boldsymbol{F}_t$。最后研究中再次使用 $\boldsymbol{F}_s$ 和 $\boldsymbol{F}_t$ 计算 $\boldsymbol{S}$ 和 $\boldsymbol{T}$ 的相似度。

CNN 模型一共包含 3 层,即:卷积层、max-pooling 层以及全连接层。以 $\boldsymbol{S}$ 为例,在卷积层中,将重点在相似通道和相异通道上设置了一组过滤器 $\{w_0,w_1\}$,用于生成一组特征,其数学运算则可写作如下形式:

$$c_{o,i}=f(w_0*\boldsymbol{S}_{[i:i+h]}^+ + w_1*\boldsymbol{S}_{[i:i+h]}^- + b_c) \tag{4}$$

其中,过滤器的大小是 $d\times h$;d 是词向量的维数;h 是窗口的大小。$A*B$ 的操作将 B 中所有元素按 A 中相应的权重进行加权求和。$\boldsymbol{S}_{[i:i+h]}^+$ 和

$\boldsymbol{S}^{-}_{[i:i+h]}$ 表示将 $\boldsymbol{S}^{+}$ 和 $\boldsymbol{S}^{-}$ 分为大小为 h 的子矩阵；b_c 是偏移项；f 是非线性函数。卷积层将会输出一组特征 $\vec{c}_o = [c_{o,0}, c_{o,1}, \cdots, c_{o,O}]$，而后将这组特征输入到 max-pooling 层中，在这一层选取 $\vec{c}_o$ 中的最大值作为输出结果，即 $c_o = \max \vec{c}_o$。如此方式，就真正解决了输入句子长短不一的问题。在 max-pooling 层中，每组过滤器生成一个一维的结果，因而最后特征向量的维数取决于过滤器的数量。如此运行后得到了句子 $\boldsymbol{S}$ 和 $\boldsymbol{T}$ 的特征向量 $\boldsymbol{F}_s$ 和 $\boldsymbol{F}_t$。

最后，研究即使用全连接层对 $\boldsymbol{F}_s$ 和 $\boldsymbol{F}_t$ 进行计算，通过 sigmoid 函数将结果归一化得到句子 $\boldsymbol{S}$ 和 $\boldsymbol{T}$ 的相似度。通过利用这种方法对候选生成句进行相似度计算，所得分值记为 S_2。

通过反复实验与观察，讨论后确定了候选生成句分值的数学运算公式可表示为：

$$Score = 0.000\ 1\ S_1 + S_2 \tag{5}$$

运算后，则按照分值高低对所得的候选生成句进行排序。

3　实验

为了对上述方法进行有效性的验证，研究采用“百度知道”的相似问句作为复述语料资源，并以目前效果较好的其它基于问句模板的复述生成方法[9]以及由 Prakash 等人[10]最新提出的基于残差 LSTM 模型的复述生成方法作为对比，进行实验。研究中，将对此阐述如下。

3.1　实验设置

研究采用哈尔滨工业大学社会计算与信息检索研究中心的语言云平台对语料进行分词、词性标注与依存关系分析。

在 RNN-LM 的使用过程中，研究以语料资源的 2/3 作为训练集，1/3 作为验证集，隐含层单元数设为 40，控制开关个数设置为 2，控制通过环反向传播错误设置为 4，词语分类为 200 类。

在使用基于相似与相异信息的 CNN 模型计算相似度的过程中，研究中对所用语料使用 Mikolov 的模型训练了 100 维的词向量，并从语料中抽取了 3 259对问题对，添加了人工标注，将其中 2 500 对作为训练集，500 对作为开发集，759 对作为测试集。设置过滤器的大小为 3，过滤器个数为 500，学习率为 0.01，共进行了 10 轮训练。

3.2　实验结果

实验过程中，研究随机选择 100 条问句作为测试集，分别用传统的模板匹配生成方法、基于残差 LSTM 的复述生成方法以及本文提出的基于模板匹配的复述生成方法进行实验，并以覆盖率与准确率作为评价指标。对此研究内容可分述如下。

（1）覆盖率。成功复述生成的问句在测试集中所占的比例，运算时可参考数学公式如下：

$$C = \frac{T_{paraphrase}}{T_{all}} \tag{6}$$

其中，$T_{paraphrase}$ 为测试集中被成功复述的问句数量，T_{all} 为测试集中的问句总数量。

如：测试集问句共 100 条，其中 70 条问句经过复述生成产生了新的结果。则其覆盖率为 70%，该指标可以反映不同方法的复述生成能力。

（2）准确率。提取的候选生成结果中正确的数量与提取的候选生成结果总数量的比值，运算时可参考数学公式如下：

$$P = \frac{N_{correct}}{N_{extract}} \tag{7}$$

其中，$N_{correct}$ 为提取的候选生成句中正确的数量，$N_{extract}$ 为提取的候选生成句总数量。

如：研究共选取了 50 条候选生成问句进行标注，其中有 20 条是正确的，则其准确率为 40%，该指标可以反映不同方法的复述生成效果。

选取候选生成结果中的 Top3 和 Top1 进行人工标注评价，实验结果见表 6。

表 6　不同方法的复述生成结果

Tab. 6　The result of different paraphrase generation methods %

	覆盖率 C	准确率 P_1（Top3）	准确率 P_2（Top1）
Template	54.0	34.6	37.0
R_LSTM	—	—	41.0
FD_Template	83.0	65.1	67.5

在表 6 中，Template 为已有效果较好的基于模板匹配的复述生成方法，R_LSTM 为基于残差 LSTM 的复述生成方法，FD_Template 为本文提出的基于模板匹配的复述生成方法。

研究分析后可知，Template 和 FD_Template 方法为基于模板匹配的复述生成方法，部分问句无法进行复述生成，因此需要统计覆盖率，且每条可复述的问句可能生成多个候选结果，因此结果中包含 Top3、Top1 两个评价部分。R_LSTM 为基于残差 LSTM 的复述生成方法，每个问句能且只能生成一种可能性最大的候选结果，因此其覆盖率无需参与对比，且不存在 Top3 结果统计。

采用本文所提出的 FD_Template 复述生成方法进行实验，并取其生成结果的 Top3，部分结果可见

表 7。其中，Q_i 为原始问句，P_j 为复述生成的候选结果。

表 7 FD_Template 复述生成的 Top3 结果

Tab. 7 Top3 results by FD_Template

Q_1：宙斯有多少个儿子
P_1：宙斯的儿子有多少个
P_2：宙斯到底有多少个儿子
P_3：宙斯的儿子有哪些
Q_2：安徒生写过什么童话
P_1：安徒生写过哪些童话啊
P_2：安徒生写过哪些童话
P_3：安徒生写过的童话有哪些啊
Q_3：产品经理是什么
P_1：产品经理是什么呀
P_2：产品经理是什么啊
P_3：到底什么是产品经理

3.3 实验分析

通过对以上结果的分析，可以得出本文所提出的 FD_Template 复述生成方法在覆盖率和准确率指标上均取得不错结果，与传统的基于模板匹配的复述生成方法和基于残差 LSTM 的复述生成方法相比，效果提升明显。

与传统的基于模板匹配的复述生成方法相比，本文的方法在模板抽取部分引入了功能词和句式精简，突出了模板的特征，增强了模板的泛化能力，探索出了一种更为适合的问句模板的表示方法；在候选生成句抽取的过程中，本文提出了一种近似于复述检测的方法—基于相似度计算与语言模型相结合的方法，该方法不但考虑了生成句的流畅性，更利用了问句所涵盖的语义信息。

与基于残差 LSTM 的复述生成方法相比，基于残差 LSTM 的复述生成方法将复述生成研究视为一种机器翻译任务。该模型对问句的语义可以进行更深程度的挖掘，但由于当前缺少大规模、高精度的复述问句语料资源，故而该模型的学习能力受到极大限制。而本文提出的方法对于语料的要求较低，在现有的低精度的复述语料资源上即可取得不错效果。

4 结束语

本文在传统的基于模板匹配的复述生成方法的基础上，加入了功能词、句式精简，可对问句模板进行更合理的表示，提出了一种新的候选生成句打分排序方法，可对候选生成句实现更为有效的抽取，且避免了基于深度学习方法开展复述生成研究时面临的大规模、高精度语料缺少的问题。通过最终的实验结果可以发现，本文提出的复述生成方法的覆盖率与准确率较高，证明其可对大部分问句进行复述并得到不错效果。

参考文献

[1] Appleby S C. Machine translation: US, US20050015240A1[P]. 2005-01-20.

[2] LIU Ting, LI Weigang, ZHANG Yu, et al. A survey on paraphrasing technology [J]. Journal of Chinese Information Processing, 2006, 20(4):25-32.

[3] LIN Dekang, PANTEL P. DIRT @ SBT@ discovery of inference rules from text[C]// Proceedings of the seventh ACM SIGKDD Conference on Knowledge Discovery and Data Mining. San Francisco, California:ACM,2001:323-328.

[4] PANG Bo, KNIGHT K, MARCU D. Syntax-based alignment of multiple translations: Extracting paraphrases and generating new sentences[J]. NAACL '03 Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1. Edmonton, Canada:ACM, 2003:102-109.

[5] ZHOU Hanyan, WU Yongyao, CHEN Jianhong, et al. A model for trans-translation[J]. Hereditas(Beijing), 2006, 28(8):1051-1054.

[6] WANG Zhiguo, MI Haitao, ITTYCHERIAH A. Sentence similarity learning by lexical decomposition and composition[J]. arXiv preprint arXiv:1602.07019,2016.

[7] MIKOLOV T, KOMBRINK S, DEORAS A, et al. RNNLM-recurrent neural network language modeling toolkit[C]//Proc. of the 2011 ASRU Workshop. Cancun:IEEE ,2011:196-201.

[8] MIKOLOV T, CHEN Kai, CORRADO G, et al. Efficient estimation of word representations in vector space [J]. arXiv preprint arXiv:1301.3781,2013.

[9] Qin Haocheng. Question paraphrase generation for question answering system[D]. Waterloo, ON, Canada: University of Waterloo, 2015.

[10] PRAKASH A, HASAN S A, LEE K, et al. Neural paraphrase generation with stacked residual LSTM Networks [J]. arXiv preprint arXiv:1610.03098,2016.