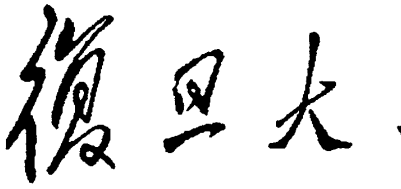

本科毕业论文

论文题目：中文开放领域问答系统训练数据扩增技术的研究

院　　系：计算机科学技术学院

专　　业：计算机科学与技术

姓　　名：许陆

学　　号：14307130056

指导教师：肖仰华

2018 年 5 月 28 日

摘要

深度学习技术近来发展迅猛，在自然语言处理领域的学术研究中，其出现的频率也越来越多。作为自然语言处理领域一个重要的研究方向，问答系统的相关研究中近年来也大量地出现了与深度学习模型有关的系统设计。深度学习模型训练时需要大量问答语料作为训练集，然而如今中文开放领域问答系统训练时普遍面临着没有足够的大规模中文开放领域数据集的问题。为了解决这一实际问题，本文提出了一个用来自动生成问答语料的问答数据扩增流水线。该流水线以少量人工标注好的问答对作为种子问答对，再利用种子问答对生成新的问答对。该流水线首先会通过一个实体替代模块，利用若干个频繁实体来替换种子问答对中的实体。其次，该流水线会通过一个 Web 搜索模块，利用 Web 信息进行第二次问答对扩增。接着，该流水线会利用机器翻译模块进行第三次问答对扩增。随后，该流水线会通过一个生成模型模块，利用一个基于长时短记忆网络构建的生成模型进行第四次问答对扩增。最终我使用上述流水线生成了一个大型中文开放领域问答系统数据集。

关键词：中文问答系统，开放领域问答系统，训练数据扩增

Abstract

Sweeping success has achieved by deep learning technology in natural language processing in recent years. As an important research direction in the field of natural language processing, the research of question answering systems has also seen a great deal of system design related to deep learning models in recent years. Deep learning model training requires a large amount of question and answer corpora as training sets. However, currently Chinese open domain question answering system field does not have a large-scale data set that can be used for training. In order to solve this practical problem, this paper proposes a question and answer corpora data augmentation pipeline for automatically generating question and answer corpora. The pipeline uses a small number of manually annotated question and answer corpora as seed question and answer corpora, then uses the seed question and answer corpora to generate new question and answer corpora. The pipeline first uses an entity replacing module, replacing the entity in the seed question-answer pair with several frequent entities. Second, the pipeline will use a Web search module to use the Web information for generating new question and answer corpora. Next, the pipeline will use a machine translation module for a third question-and-answer pair amplification. Subsequently, the pipeline will use a generative model module to generate a fourth question and answer corpora augmentation using a generative model based on LSTM memory networks. In the end, the model of this paper generates a large question and answer corpora of the Chinese open domain question answering system.

Keywords: Chinese question answering system, open domain question answering system, training data amplification.

目录

图目录

表目录

第一章 绪论

1.1 研究背景和意义

问答系统，是指能准确、简洁地回答用户以自然语言形式提出的问题的系统。中文开放领域问答系统的特性是：它能够回答不限定领域的中文自然语言问题，其面对的任务不会限定在一个或者若干个特定领域中。问答系统因其重要的应用价值多年来一直得到了学术界和工业界的广泛关注。近年来，各种深度学习模型被提出并成功地应用于问答系统中[1][2]。在实现最先进的性能的同时，这些模型依赖于大量的标记数据。在训练中文问答系统中的深度学习模型时，研究者们遇到的主要障碍往往是缺乏标签数据。但是，收集大型问答数据集非常困难，人工标记问答对的过程在实践中昂贵而又耗时。现在学术界公开的大型中文开放领域问答数据集很少，如 NPLCC-2016、百度 Webqa[3]等。这些数据集所包含的问题数量远远无法满足训练中文开放领域问答系统所需要的数据需求。这样的现状阻碍了中文开放领域问答系统的发展与应用。与人工获得带标签的问题答案对所付出的代价相比，搜索互联网数据的成本是微不足道的。在本文的工作中，我主要研究了以下两个问题：在已有少量中文问答语料的前提下，是否可以利用互联网的信息，来自动扩增出可供中文开放领域问答系统训练使用的问答对？是否可以通过训练一个生成模型来完成相似的任务？这两个问题具有挑战性，因为互联网上的数据大多是未标记的非结构化数据，从中自动提取出高质量的问题以及找到与之匹配的答案是有一定难度的；对于利用生成模型自动产生问答对这个任务而言，我也必须找到一种方法来衡量通过生成模型所生成的问答对的质量。当然，这个问题对于工业界和学术界的中文开放领域问答系统构建都具有重要的实用价值。足够的数据可以让问答系统中的深度学习模型学到复杂而灵活的自然语言表达方式，从而能够在保证问题系统的训练质量与回答能力方面起到关键的作用。

1.2 相关研究综述

问答系统训练数据的生成是近年来一个十分热门的研究方向。QALD[4]和 FREE917[5]分别提供了包含数百个人工标注的问题，但是这样的数据集对于深度学习训练来说是远远不够的。Olney 等的工作[6]将知识图谱中的三元组作为输入，以其中的谓词关系定义问题模板，并使用三元组中的实体替换所选问题模板中的占位符标记。借着同样的思路，Duma 等[7]通过使用由关系定义的模板并相应地替换主体和对象的占位符标记来从三元组生成简短的问答对。Berant 和 Liang[8]的工作解决了确定性地生成一组候选逻辑形式的问题，其中每个逻辑形式都有自然语言的规范实现。Wang 等[9]针对“篮球”等特定领域生成问答对。他

们首先利用知识图谱中的三元组生成逻辑表达式，然后通过众包的方式将逻辑表达式转化为问题。这种方法可以生成高质量的问题对，然而利用人工来标记数据的代价十分昂贵，Yih 等[10]也发现，通过众包收集的 WEBQUESTIONS 答案中只有 66%完全正确。Su 等[11]阐述了构造一个富含特征的数据集的过程。尽管他们对其生成的数据集进行了十分详尽的分析，但是他们的问题也是使用了众包平台 Mturk 人工生成的。Serban 等[12]将产生新的问答对的过程视为将三元组翻译成问题的过程。其基于了机器翻译的视角来看待问答数据扩增的思路给予了我一定的启发。Seyler 等[13]利用了维基百科页面中的相互引用数量来判断一个实体的热度，然后对于实体的各个谓词构建了问题模板来生成新问题。问题模板的做法减少了人工构造数据的工作量。然而，构造问题模板的过程依然是需要人工投入的过程，该系统所能构造的问答对也因为问题模板的有限性而有所限制。因此，我的研究使用了 Web 信息和 BLEU 评价标准[17]来自动地生成问答对并检查问答对的质量，针对过去问答系统数据构造中所遇到的人工构造成本高、生成数据质量低等问题提出了针对性的解决方案。

1.3 研究内容与文章组织架构

我提出了一个流水线框架来实现整个中文开放领域问答语料数据扩增的过程。该框架的出发点是少量人工标注的中文问答语料。利用这些初始语料，我会通过以下四个模块来进行数据扩增：1、实体替代模块：我会使用若干个知识图谱中的实体替代起始问题中的实体，得到若干个新问题。同时，利用知识图谱中的相应属性信息来获得新问题的答案，从而获得新的问答对。2、Web 搜索模块：该模块会通过在搜索引擎中对于问题进行搜索来扩增问答对，利用搜索引擎所给出的结果，该模块能够自动得到并鉴别可能的新问答对。3、机器翻译模块：通过利用机器翻译系统在不同语言之间来回翻译的方式来扩增问答对。4、生成模型模块：我使用了深度学习的方法训练了一个生成模型来扩增问题。我利用 NLPCC-2016 数据集作为原始的问答对进行扩展。实验结果表明，我的流水线框架显著地丰富了源数据集的问题表达方式。更具体地说，通过我的流水线框架之后，平均每输入一个原始问答对可以生成 17.3 种新的问答对，而这些新问答对的质量利用 BLEU 指标评估均超过了 0.15，体现了这些通过扩增得到的问答对的合理性。我们的贡献有四重。首先，与以往大多数关于通过构造模板的方式来人工构建数据集的研究不同，我研究了一个重要又有挑战性的问题，即自动生成中文问答系统训练问答对的方法。其次，我提出了一个可迭代的流水线框架，利用这个框架可以利用少量的问答对扩增出大量的问答对。第三，我介绍了一个可以用来衡量自动生成的问答语料质量的方法。第四，我的实验验证了在中文问答

对数据集上，使用 Web 信息或生成模型来扩增问答对数据集的可行性。

第二章 相关技术概述

2.1 知识图谱

在整个流水线中，我多次使用了知识图谱作为结构化数据来源。知识图谱以三元组形式存储关于实体和关系知识的。知识库也可以自然地表示为有向图，实体是图中节点，属性是图中的有向边。如图 1 所示，在大型中文知识图谱 CN-DBpedia[14]中，复旦大学是一个实体。这个实体包括多个属性，在图中以有向图的形式标注出来。本文中使用 CN-DBpedia 作为中文知识图谱来源。它基于百度百科等互联网信息构建，拥有 16,869,222 个实体和 222,922,390 个关系。

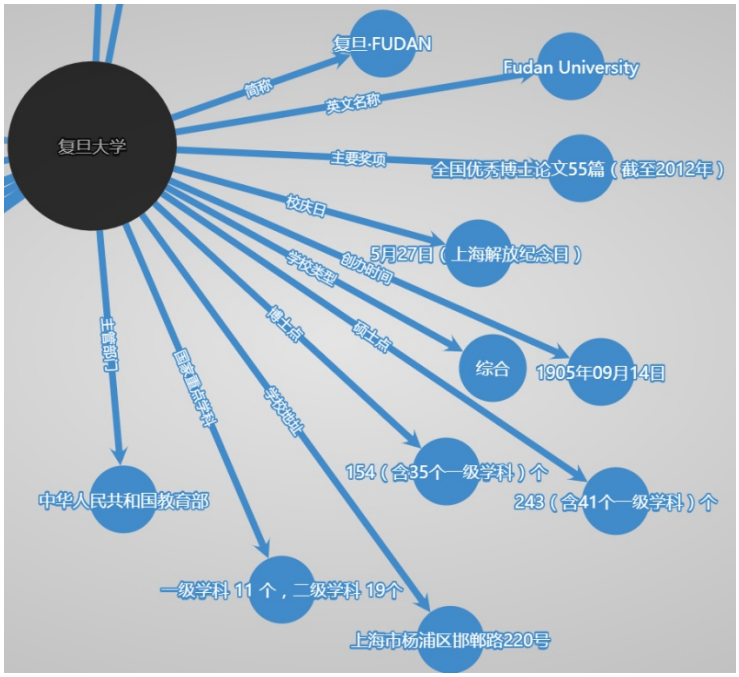

图 1：大型中文知识图谱 CN-Dbpedia

2.2 Google 翻译

Google 翻译[15]是一个完全由深度学习网络构造的翻译系统(GNMT)。该系统使用循环神经网络（RNN）来直接学习一种语言的语料（如一个句子）到另一种语言的语料（如另一个句子）的映射。神经机器翻译（NMT）会把整个输入的语料以句子为单位视作若干个整体，作为机器翻译任务的基本单元。这种方法相比之前流行的基于短语的机器翻译系统的优点在于，神经机器翻译的方法所需的工程量少得多。同时其翻译效果较基于短语的机器翻译系统也有明显提升。本文中使用了 google 翻译来实现流水线中的机器翻译模块。

第三章 问答数据扩增流水线

3.1 流水线模型

如图 2 所示，针对扩增中文开放领域问答系统数据集这一任务，我提出了一个流水线模型。利用初始的中文问答对语料，我会通过以下四个模块来进行数据扩增：实体替代模块、 Web 搜索模块、机器翻译模块、生成模型模块。对于每个模块的实现细节将在下面的篇章中详细介绍。值得注意的是，我在每个模块的实现中都应用了特定的评估方式，以对生成的问题或者问题中的实体进行了限定，从而保证了生成的新问答对的质量。

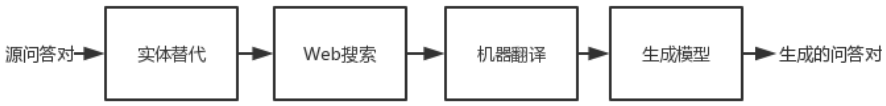

图 2：流水线框架

3.2 实体替代模块

对于原始问答对中的问题 q 及其答案 a，我使用了 xie 等[16]的实体链接技术对其进行了实体链接，得到了 q 中的实体表示 e。在 CN-Dbpedia 中，我会使用这个实体 e 进行搜索，得到 e 所拥有的所有属性集合 R 及各属性对应的属性值集合 O。如果我发现答案 a 在集合 O 中，则我便可以得到原始问题在对于知识图谱中的哪一个属性 r 进行提问。通过这个步骤，我得到了问题 q 中的实体 e、属性 r 和答案 a。例如：对于“问题：中国的首都是哪里？答案：北京”这个原始问答对。我通过实体链接得到了原问题中的实体“中国”。再用“中国”在知识图谱中进行搜索发现“中国”这个实体有且仅有一个属性的属性值是“北京”。这个属性就是“首都”。通过这个流程，我就得到了原问题的实体“北京”和问题属性“首都”。接着，该模块会试图通过替换原问题中的实体的方式对于原问题进行扩增。形式化地说，对于一个问题 q，在找到其实体 e、属性 r、答案 a 之后，该模块会在知识图谱中搜索所有包含属性 r 的实体，得到一个实体集合 E。在这个集合中，我挑选了 K 个实体用来做实体替换。挑选的标准是被选中的实体所包含的属性数量一定要超过一个阈值 M。这就保证了我所选中的实体是相对频繁的实体。例如“问题：中国的首都是哪里？答案：北京”。在已知其实体是“中国”、属性是“首都”的情况下，我会在知识图谱中搜索所有包含“首都”这个属性的实体，在搜索时设置实体必须包含超过 M 个属性、并在找到 K 个符合条件的实体后停止。这样我

就得到了 K 个包含“首都”这个属性的实体。最后，对于原问题 q 和其中的实体 e，我用这 K 个实体替换掉原问题中的实体 e。这样我就生成了 K 个新的问题。由于知识图谱的存在，我也可以很容易地搜得这 K 个新问题所对应的答案。

3.3 Web 搜索模块

在这个模块中，我会将初始问题和实体替代模块所扩增的问题合并输入，在互联网搜索引擎“百度搜索”中使用问题作为关键词进行搜索。如果百度后台能够识别该问题并利用百度百科的结构化信息来回答该问题，则会在百度返回的结果页上显示一个特定的答案框，如图 3 所示，这就是搜索关键词是否是一个自然语言问题的判断标准。如果某个关键词的搜索结果页中不包含答案框，则将其判定为非自然语言问题。据此，我这样设计 Web 搜索模块：首先，对于这个模块的输入问题，我均会用作关键词进行百度搜索。其次，在每次利用关键词 q 搜索后，如果关键词 q 能够被百度作为问题识别并回答，则利用百度搜索结果页中的“相关搜索”结果，获得关键词 q 的相关搜索集合 QS，如图 4 所示。将所有的相关搜索选项都作为新的关键词进行搜索，利用上文中的判断标准可以得到所有相关搜索选项中是一个自然语言问题的选项。在图中我们可以看到该问题的答案也出现在了搜索结果页面中。这些在相关搜索选项中找到的新问题和答案就是这个模型生成的新问答对。同时这个模型可以不断迭代，利用广度优先的搜索方法得到多层搜索结果，从而获得更多的新问题。即：QS 中被判断为是一个自然语言问题的元素 qs 可以再被用作关键词进行一次百度搜索，得到搜索结果中的相关搜索选项集合 QSS。QSS 中又可以找到新的自然语言问题。如图 5 所示，整个搜索的过程构成了一颗搜索树，当一个节点被判定为是自然语言问题后，它的相关搜索子节点便会被用来作为关键词进行下一层的搜索。如果一个节点被判定为非自然语言问题，它的相关搜索选项将不会被用来进一步搜索。出于为了防止模型的迭代陷入循环和提高搜索效率的目的，我设置了一个迭代层级的阈值 C。在本文的实验中，C 设置为了 4。即对于任意一个输入的问题 q，递归地搜索其相关搜索选项中是否包含自然语言问题的过程至多进行 4 层。

图 3：使用百度搜索搜索原问题

图 4：百度相关搜索结果

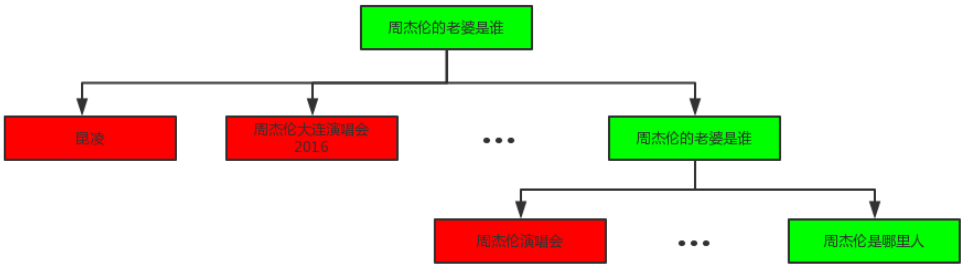

图 5：广度搜索树(绿色的节点被判定为自然语言问题)

3.4 机器翻译模块

在这一模块中，我会将所有之前模块生成的新问题作为输入。利用谷歌翻译系统，我先将输入翻译成任一外语 E，再将其翻译回来。通过这样的方式，我得到了原问题的新表述方式。新问题的答案是和原问题一样的。为了评价通过机器翻译模块生成的问答对质量，我引用了机器翻译中的 BLEU 评价标准[17]。有关 BLEU 的技术细节将在下文中详细讨论。在 1-gram 以及 2-gram 的维度上，我设

立了阈值 B。如果机器翻译模块所输出的新问题与原问题计算 BLEU 得到的分数低于 B，则认为生成的新问题是一个低质量问题、并将其丢弃。在我的实验中，B 值设为了 0.15。

BLEU 是一个机器翻译质量的评价标准。BLEU 标准通过比较生成的新问题与原问题的相似度、新问题的长度等指标来评对新问题的质量进行打分。其公式如下：

$$\mathrm{BP} = \begin{cases} 1 & if\ c > r \\ e^{(1-r/c)} & if\ c \le r \end{cases}$$

$$\mathrm{BLEU} = \mathrm{BP} \cdot \left(\sum_{n=1}^{N} w_n \log p_n\right)$$

其中 r 是标准答案的长度，c 是被评估的翻译的长度，N 代表了 N-gram，Wn 代表了每个 n-garm 的权重。Pn 是指被评估翻译语句里面的 n-gram 在所有标准答案语句里面出现的概率。BLEU 公式奖励了与原句相似的翻译并惩罚了过短的翻译结果。

3.5 生成模型模块

在这一模块中，我同样以之前的模块所生成的全部问答对数据作为模块的输入。我训练了一个生成模型来实现生成新问答对的任务，如图 6 所示。这个生成模型包含了两个组成部分：一个编码器(encoder)和一个解码器(decoder)。这个生成模型的结构如图 X 所示。

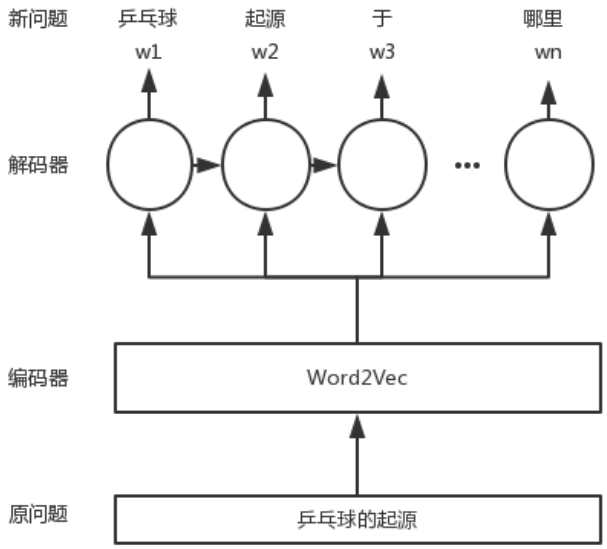

图 6：生成模型

3.5.1 编码器

对于输入的问句，这个编码器试图将中文自然语言的句子编码为向量来提供给解码器训练。在使用结巴分词对于问句分词以后，我会使用训练好的 word2vec 模型[18]将词语编码成词向量。word2vec 模型的优点在于它可以使得词意相近的两个词在被编码后产生的两个向量的欧氏距离较小。这一步得到的输出即为解码器的输入。

3.5.2 解码器

解码器是由一层长时短记忆网络(LSTM)[19]构成的。该网络的输入是编码器对于问句进行编码后得到的词向量。通过解码器，输入的词向量转化为输出，这个输出就是新生成的问题，而新问题的答案和原问题是一样的。LSTM 网络能够对句子中词语之间的相互依赖关系进行建模，并将词语的顺序考虑在内。LSTM 网络由一组存储单元组成。每个存储单元以编码器产生的词向量和之前的一个存储单元的输出作为该存储单元的输入。每个存储单元有四个基本的元素：一个输入门，一个遗忘门，一个记忆状态和一个输出门。在本模型中，遗忘门首先从编码器和前一个存储单元的的输出门得到输入，并决定从两个数据中弃掉哪个数据。接着输入门会决定更新哪一个值。记忆状态会存储被更新的值。最终，输出门会决定输出什么。在本模型中，我使用了 Zaremba 等[19]提出的 LSTM 版本。存储单元计算时涉及到的公式如下：

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i)$$
$$j_t = \sigma(W_{xj}x_t + W_{hj}h_{t-1} + b_j)$$
$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + b_f)$$
$$o_t = \tanh(W_{xo}x_t + W_{ho}h_{t-1} + b_o)$$
$$c_t = c_{t-1} \odot f_t + i_t \odot j_t$$
$$h_t = \tanh(c_t) \odot o_t$$

在公式中，σ 函数是 sigmoid 函数，i,f,o,c 分别代表输入门、遗忘门、输出门和记忆状态单元激活向量。j 是为了计算 c 值用到的中间变量。W，b 都是 LSTM 网络的超参数。

在实际训练生成模型模块的过程中，对于编码器，我训练了一个 300 维的 word2vec 模型，将每一个中文词语映射到了一个 300 维的向量空间中。对于解码器，我使用了 128 个 LSTM 节点构成的单层 LSTM 网络。我选用了 Categorical_Crossentropy 算法作为我模型的损失函数、Adam 方法作为模型的优化方法、Softmax 算法作为了 LSTM 节点的激活函数。模型调参时，批规模设置

成了 20，训练循环次数设置为了 200，学习率设为了 0.001。另外，为了避免模型过拟合，我将模型的剪枝系数设为了 0.5。我使用了与机器翻译模块相同的评价标准来评价生成模型所得结果的质量，删去了一部分生成出来的新问题，保留了所有 BLEU 分高于 0.15 的新问题。

第四章 实验结果

4.1 问答对生成情况

关于整个流水线中 Web 搜索模块搜索新问题的算法、评价通过机器翻译模块、生成模型模块自动生成得到的新问题的方式、生成模型的模型架构等细节，已经在第三章中得到了详细的讨论。本章介绍我使用了这一套流水线来生成新问题的实验结果。我在 NLPCC-2016 数据集上针对我的模型进行了实验。在使用了 Chen 等人(引用)提出的中文实体链接模型后，从中随机抽出了 406 句句子作为我整个流水线的初始问答对。在实体替代模块中，我将新实体的属性个数下限设为了 20 个，生成新实体的数量上限设定为了 100。通过在存有中文 CN-Dbpedia 知识图谱的 SQLServer 中进行 SQL 搜索，我得到了 40105 个新问题。新问题的示例如图 8 所示。接着我通过 Web 搜索模块，对于之前所获得的全部问题进行基于深度优先的 Web 搜索。在我设置了最多递归层数为 4 的情况下，一共获得了 53830 个新问题。我将到这一步为止所有的问题聚合起来，统计了其中所包含的不同的问法数，统计的结果为这 9 万多个新问题种包含了 6310 种不同的问法。我将这 6310 种问法中的实词用占位符来填充，以防止生僻实词对于翻译效果的影响。接着将所有的问法输入到了机器翻译模块，先全部翻译为英文，再全部翻译回中文。一共获得了 6310 条译文。对于这些译文，我使用了 BLEU 评测的方式评价其质量。我使用了结巴分词与 NLTK 数据包[20]来计算每一条译文的 BLEU 分数，并且只保留分数大于 0.15 的译文。结果有 1328 条译文被选了出来，这些译文中包括了 470 种新的问法。在合并了所有的问法后，我将这 6780 条新问法作为生成模型模块的输入和训练语料。经过训练，生成模型模块产生了 622 条合格的新问题，其中包含了 239 种新的问法。表格 1~表格 5 展现了这整个新问答对生成过程的实际效果。

为了验证该流水线生成的问答对对于中文开放领域问答系统的意义，我基于复旦大学图数据管理实验室的中文问答系统进行了实验。该实验以 NLPCC 数据集中的 406 个问题作为测试集。如表格 6 所示，在引入了扩增的问答对作为额外的特征以后，该中文问答系统回答 NLPCC 中 406 个问题的准确率提升了 1.2%。这一实验结果证明了利用本文所提出的的流水线产生的新问答对，可以一定程度上提高中文问答系统的回答准确度。

	实体替代模块	Web 搜索模块	机器翻译模块	生成模型模块
新问题数	40105	53830	1328	622
新问法数	0	6310	470	239

表格 1：各个模块生成的新问题和新问法数量

模型输入	模型输出
陈佳属于哪个星座？	JJ 属于哪个星座？
	姚明属于哪个星座？
	郑源属于哪个星座？
	赵薇属于哪个星座？

表格 2：实体替代模块输出示例

模型输入	模型输出
京剧起源于哪个国家？	京剧的起源简介
	武术起源于哪里
	京剧的由来
	足球起源于哪个国家

表格 3：Web 搜索模块输出示例

模型输入	模型输出
#实体#起源于哪个国家	#实体#来自哪个国家
#实体#有没有危险啊？	#实体#有危险吗？
#实体#内设了什么机构？	#实体#有什么样的组织？
请问#实体#的董事长是谁	谁是#实体#的董事长

表格 4：机器翻译模块输出示例

模型输入	模型输出
#实体#有哪些天敌	#实体#的天敌是什么
#实体#在哪里出版	#实体#在哪里发布
#实体#使用的方式	#实体#使用的方法
#实体#当前版本号是多少	#实体#现在版本号是什么

表格 5：生成模型模块输出示例

	未使用生成问答对作为额外信息时中文问答系统的准确率	使用了生成问答对作为额外信息时中文问答系统的准确率
准确率	48.9%	50.1%

表格 6：新生成的生成问答对对于提高中文问答系统准确率的效果

4.2 模块必要性评估

为了验证整个流水线设计的合理性，我又分别实验了从流水线中移除实体替代模块后 Web 搜索模块的生成结果和从流水线中移除 Web 搜索模块后机器翻译模块的生成结果，实验结果如表格 6~表格 7 所示。实验结果充分证明了这个流线的各个模块的存在意义：而如果不使用实体替代模块对于原来的 406 条 NLPCC 问题进行频繁实体替换，那么 Web 搜索时所涉及到实体中不频繁的冷门实体会占有很大比例。这样 Web 搜索时的效率就会大为下降。机器翻译模块与生成模型模块理论上对于一种源问法只能扩展一种新问法。因此为了使得后两个模块有效率，在其前面的 Web 搜索模块所提供的大量问法就尤为重要。更何况，脱离了 Web 搜索模块所提供的大量问法，生成模型模块也无法得到训练 LSTM 模型所需的数据集。因此，整个流水线中的各个模块均有其存在的重要价值，同时模块间的先后顺序也具备了一定的合理性。

	有实体替代模块	无实体替代模块
Web 搜索模块平均每个输入问题能产生的新问题数	1.33	0.45
Web 搜索模块平均每个输入问题能产生的新问法数	0.16	0.05

表格 7：有无实体替代模块对于 Web 搜索模块的影响

	有 Web 搜索模块	无 Web 搜索模块
机器翻译模块平均每个输入问题能产生的新问题数	0.21	0.03
机器翻译模块平均每个输入问题能产生的新问法数	0.07	<0.01

表格 8：有无 Web 搜索模块对于机器翻译模块的影响

4.3 错误分析

实验过程中生成的新问答对中有质量较差的问答对。本章试分析产生了这些问答对的原因。在实体替代模块部分，理论上生成的问题均为可用的问答对。但是实际生成的问答对中有相当一部分的问题中的实词是热度较低的。这影响了之后各个模块生成问题的效率。这一问题的根源是因为知识图谱中没有与实体热度相关的信息，使用实体的属性数量是一个不完全准确的替代方案。在实际实验的

过程中，我以拥有超过 20 个属性的实体作为热门实体的判断标准。但是这样的标准依旧未筛去一些冷门实体，如图 7 所示。在 web 搜索模块中，出于搜索性能的考虑，我限制了在搜索引擎中广度搜索的层数，这也使得这一模块中生成的新问题的多样性受到了限制。另外，一些新问答对因为未能被搜索引擎判定为自然语言问题而未被该模块记录下来。在机器翻译模块与生成模型模块中，模型生成的问题中均有一部分错误。机器翻译模块中在不同语言之间来回翻译的过程和生成模型模块中对语言编码和解码的步骤均不可避免地会造成信息的丢失和扭曲。这样的丢失和扭曲是导致模型生成的结果中出现错误的主因。

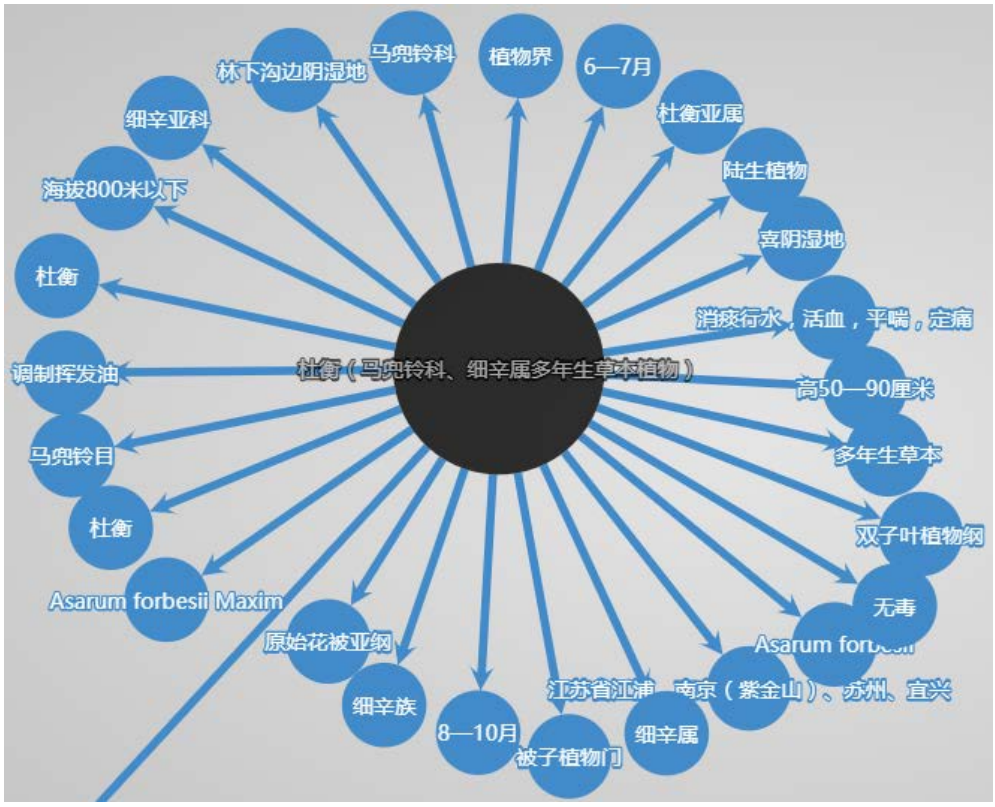

图 7：冷门实体“杜衡”

第五章 总结与讨论

我提出了利用少量中文开放领域问答对扩增得到大量问答对的一整套流水线框架。使用这个综合了知识图谱信息、Web 信息与生成模型的框架，我从 406 个人工标注的中文问答对出发，得到了超过 5000 种完全不同的问法。这个实验结果充分证明了本文提出的方法几种扩增技术对于解决扩增中文开放领域问答语料这一挑战有着良好的效果。

同时，在研究完成本文的过程中，有一个任务我认为值得在未来进行更加深入的研究：实体表示方法扩增。利用本文的框架，我们可以对于同一个问题谓词得到多样化的表示(即“姚明什么时候出生的？”所对应的“出生日期”)。然而，对于同一个问题实体(即上例中的“姚明”)，如果能够自动生成一些多样化的表示(例如“大姚”、“小巨人”等等)，将大大提高自动生成问题的多样性。在这个问题上，英文的研究已经取得了一些进展[11], 然而由于在中文实体链接标注语料方面的缺乏等原因，这任务在中文语料上有较大的实现难度。期待中文自然语言处理学术界在这一问题上投入更多的关注，在未来能够出现有效解决这一问题的方案。

参考文献

[1] B. Dhingra, H. Liu, Z.Yang, W. W. Cohen, and R.Salakhutdinov. Gated-attention readers for text comprehension. *arXiv preprint arXiv: 1606.01549*, 2016. 7

[2] C. Xiong, V. Zhong, and R.Socher. Dynamic coattention networks for question answering. *arXiv preprint arXiv:1611.01604*, 2016. 7

[3] P. Li, W. Li, Z. He, X. Wang, Y. Cao, J. Zhou, and W. Xu. Dataset and neural recurrent sequence labeling model for open-domain factoid question answering. *arXiv preprint arXiv: 1607.06275*, 2016. 7

[4] V. Lopez, C. Unger, P. Cimiano, and E. Motta. Evaluating question answering over linked data. *Web Semantics: Science, Services and Agents on the World Wide Web, 21:3–13*, 2013. 7

[5] Q. Cai and A. Yates. Large-scale semantic parsing via schema matching and lexicon extension. In *Proceedings of ACL*, 2013. 7

[6] A.M. Olney, A. C. Graesser, and N. K. Person. Question generation from concept maps. D*ialogue and Discourse, 3(2):75–99*, 2012. 7

[7] D. Duma and E. Klein. Generating natural language from linked data: Unsupervised template extraction. *ACL, pages 83– 94*, 2013. 7

[8] J. Berant and P. Liang. Semantic parsing via paraphrasing. In *Proceedings of ACL, volume 7, pages 1415–1425*, 2014. 7

[9] Y. Wang, J. Berant, and P. Liang. Building a semantic parser overnight. In *Proceedings of ACL*, 2015. 7

[10] W. -T. Yih, M. Richardson, C. Meek, M. -W. Chang, and J. Suh. The value of semantic parse labeling for knowledge base question answering. In *Proceedings of ACL*, 2016. 8

[11] Y. Su, H. Sun, B. Sadler, M. Srivatsa, I. Gur, Z. Yan, and X. Yan. On generating characteristic-rich question sets for QA evaluation. *Conference on Empirical Methods in Natural Language Processing, pages 562-572*, 2016. 8, 21

[12] I. V. Serban, A. Garcia-Duran, C. Gulcehre, Sungjin. Ahn, S. Chandar, A. Courville, and Y.Bengio. Generating factoid questions with recurrent netural networks: the 30m factoid question-answer corpus. *arXiv preprint arXiv:1603.06807*, 2016. 8

[13] D. Seyler, M. Yahya, and K. Berberich. Gemerating quiz questions from knowledge graphs. *International Conference on World Wide Web, pages 113-114*, 2015. 8

[14] B. Xu, Y. Xu, J. Liang, C. Xie, B. Liang, W. Cui and Y. Xiao. CN-DBpedia: A never-ending chinese knowledge extraction system. *IEA/AIE pages 428-438*, 2017. 10

[15] Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao. K. Macherey, J. Klingner, A. Shah, M. Johnson, X. Liu, L. Kaiser, S. Gouws, Y. Kato, T. Kudo,

H. Kazawa, K. Stevens, G. Kurian, N. Patil, W. Wang, C. Young, J. Smith, J. Riesa. A. Rudnick, O. Vinyals, G. Corrado, M. Hughes, and J. Dean. Google's neural machine translation system: bridging the gap between human and machine translation. *arXiv preprint arXiv: 1609.08144*, 2016. 10

[16] C. Xie, J. Liang, L. Chen, Y. Xiao, H. Tong, K. Zhang, H. Wang and W. Wang. Automatic Navbox Generation by Interpretable Clustering over Linked Entities. *ACM DOI: 10.1145*, 2017. 11

[17] K. Papineni, S. Roukos, T. Ward, and W. -J. Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on ACL, pages 311–318*, 2002. 7, 13

[18] T. Milolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of world representations in vector space. *arXiv preprint arXiv: 1301.3781*, 2013. 15

[19] W. Zaremba, I. Sutskever, and O. Vinyals. Recurrent neural network regularization. *arXiv preprint arXiv: 1409.2329*, 2014. 13, 15

[20] E. Loper, and S. Bird. NLTK: The Natural Language Toolkit. *arXiv preprint arXiv: cs/0205028*, 2002. 17

致谢

自从 2016 年初加入肖仰华老师的实验室以来，我在学术与科研上得到了肖老师大量悉心指导。肖老师以他对于知识图谱、自然语言处理、图数据挖掘、机器学习领域渊博的学识与充分的实际应用经验，给予了我巨大的帮助。在我对于研究方向方法出现迷茫的时候，与肖老师的交流总能给我重要的启迪，指引我前进的方向。在此，我要向肖老师表达我发自内心的感谢！

同时，我要感谢学校和院系，尤其是我在复旦大学计算机科学与技术系本科阶段的四年学习过程中所有教导过我的老师们。在这四年中，我对于计算机科学中的各个研究领域有了广泛的认识。这离不开各位老师的指导和关心。我还要感谢我的四年同窗，尤其是程君同、梁建泽等热心同学。他们作为从高中就参加信息学竞赛的同学，在计算机编程与研究方面已经颇有建树。在这几年里他们在我完成计算机相关任务时给了我许多帮助，加速了我的成长。

我也要感谢我的家人。他们四年以来对我经济和精神上的支持使我能够专心完成学业，不被生活中的困难所干扰。他们对我的爱一直是我前进道路上的巨大动力。

许　陆

2018 年 5 月 28 日