硕士学位论文

基于问句实体扩展和全局规划的答案摘要方法研究

RESEARCH ON ANSWER SUMMARIZATION WITH QUESTION ENTITY EXPANSION AND GLOBAL INFERENCE

赵惜墨

哈尔滨工业大学

2015 年 6 月

国内图书分类号：TP391.3　　学校代码：10213
国际图书分类号：681.37　　密级：公开

工程硕士学位论文

基于问句实体扩展和全局规划的答案摘要方法研究

硕 士 研 究 生：赵惜墨

导　　师：王晓龙教授

申 请 学 位：工程硕士

学　　科：计算机技术

所 在 单 位：计算机科学与技术学院

答 辩 日 期：2015 年 6 月

授予学位单位：哈尔滨工业大学

Classified Index: TP391.3
U.D.C: 681.37

Dissertation for the Master Degree in Engineering

RESEARCH ON ANSWER SUMMARIZATION WITH QUESTION ENTITY EXPANSION AND GLOBAL INFERENCE

Candidate: Zhao Ximo
Supervisor: Prof. Wang Xiaolong
Academic Degree Applied for: Master of Engineering
Speciality: Computer Science and Technology
Affiliation: School of Computer Science and Technology
Date of Defence: June, 2015
Degree-Conferring-Institution: Harbin Institute of Technology

摘 要

随着互联网的普及，人们在网络上寻求帮助的需求越来越迫切。以 Yahoo answers、百度知道为首的社区问答系统(Community Question Answer，CQA)变得越来越重要。相较于传统的问答系统和 FAQ 问答，CQA 系统的问题规模更大、问题种类更多，极大地丰富了互联网的资源，为人们在互联网上寻找信息提供了便利。但是 CQA 中单个答案存在着不全面、与问题不相关等问题，使得 CQA 系统中问答对质量下降，影响用户满意度。答案摘要方法是解决此类问题的一种途径，通过对同一问题下的所有答案进行摘要，得到一个全面、与问题相关的答案摘要。本文从问句出发，利用知识图谱对问句进行实体扩展，找到和问句有关联的句子，然后利用整数规划的全局算法进行答案摘要。本文研究内容主要有三方面：

1. 研究基于知识图谱的问句实体扩展问题。单纯从问题很难预测答案内容。本文的切入点是利用知识图谱的现有知识，从问题预测答案的内容。采用基于知识图谱的问句实体扩展方法，先找出问题的实体，根据实体的关系对问题中的实体进行扩展。这些实体有可能在答案中出现。本文对这些实体的权重进行估计。实验表明，根据扩展出的实体可以准确的预测出答案中的实体。因而扩展的实体即是答案的内容。

2. 研究如何用问句实体扩展的实体得到一个与问句相关、完整的答案摘要。使用句子压缩和句子过滤两种手段，过滤掉信息含量低的答案句子。对于答案句子只保留扩展的实体，并设定一个在规定字数下扩展实体权重最大化的目标函数。通过整数规划算法可以求解目标函数，得到答案摘要。增加了答案句子质量评估和未命中实体权重估计两种特征，优化了答案摘要算法。

3. 实现了一个基于本文算法的答案摘要系统。系统有问题检索和答案摘要两个模块：问题检索模块检索与用户输入问题最相近的 10 个问题，并加以展示；摘要模块对用户选择的问题进行摘要，展示问题描述，最佳答案及问题下所有的答案。

关键词：答案摘要；知识图谱；实体扩展；全局规划

Abstract

With the popularity of the Internet，the need for people to seek help is becoming urgent. CQA service such as Yahoo！Answer and Baidu Zhidao is becoming more important. Compared to traditional Question & Answering System and FAQ System， CQA system has a larger scale and more categories, greatly improves the resources of Internet，brings convenience to people for seeking information. But each answers in CQA may be incomplete or irrelevant to questions，this causes the quality degradation of question and answers pairs，reDUCing users' satisfaction. Answer summarization is a solution to this problem. By summarizing all the sentences of the question， we can get a complete and relevant summary. This paper goes from question，takes use of knowledge graph to do entity expansion，finds all the sentences relevant to the question and finally gets the summary by mixed integer linear programming algorithm。This paper includes three aspects as follows：

1. Research on question entity expansion based on knowledge. It's difficult to predict answer's content just based on question. Our intuition is that knowledge already has huge amounts of knowledge, we firstly find question's entities, then expand question's entities based on entities' relations which may appear in the true answer. We estimate the weight of these entities. As a result, the expanded entities can accurately predict answer's entities, this tells the expanded is question's content.

2. Research on the algorithm to get a complete, relevant summary by using the expanded entities. We use sentence compression and sentence filtering to remove the poor information sentence. For the remained sentences, we only consider the expanded entities, and build an object function. By maximizing the objective function, a readable summary is obtained. The sentence quality and missing entity weight estimation is used to optimize the algorithm.

3. We built a CQA answer summarization system, which also contains question retrieval. The question retrieval module can get the closest question to user typed. Then the system display best answer and answer summarization together.

Keywords: answer summarization，knowledge graph，entity expansion，global inference

目　录

第 1 章 绪　论

1.1 课题背景

随着互联网的普及，人们在网络上寻求帮助的需求越来越迫切。以 Yahoo answers[1]、百度知道[2]为首的社区问答系统(Community Question Answer，CQA)变得越来越重要。相较于传统的问答系统[1]和 FAQ 问答系统[2]，CQA 系统中的问题规模更大、问题种类更多，极大地丰富了互联网的资源，为人们在互联网上寻找信息提供了便利。

CQA 服务的流程如下。在 CQA 系统上，用户提出一个问题，该社区知道答案的用户对该问题进行回答。在经过一段时间后，最佳答案被提问者自己或社区以投票的方式选出。这些答案和问题对被 CQA 系统存储和索引，可以在其他用户检索相似问题时重用。在理想的情况下，对于用户新提出的问题，搜索引擎在 CQA 系统中搜索相似的问题，给提问者提供该问题的可能答案。然而，每个用户的答案内容具有高噪声、不全面等问题，每个问题的回答中可能产生高水平的答案，也可能有很多广告信息或者不相关信息，即使对于最佳答案来说，也存在着这样的情况。这些问题减弱了 CQA 系统的使用体验和用户的满意程度。

答案摘要是解决 CQA 系统中单个答案不全面、水平不一致、高噪声问题的一种有效手段。答案摘要通过聚合所有答案，力图给提问者提供一个完整、与问题相关、全面的摘要。答案摘要可以提高 CQA 中问答对的质量，可以提高 CQA 的服务和用户满意度。

1.2 课题研究的目的和意义

随着人们在互联网上寻求帮助的需求越来越多，CQA 系统在人们生活中所占的位置越来越高，因而解决 CQA 系统中的答案不全面、水平不一致、高噪声的问题有着非常重要的实际意义。一个可行的解决办法是，对这些答案进行摘要，从这些答案中选出与问题相关且又可以互相补充的句子，这类方法称为答案摘要。

随着互联网信息越来越多，知识图谱的地位显得越来越重要。知识图谱内

[1] https://answers.yahoo.com/
[2] http://zhidao.baidu.com/

存储着现实世界中实体和实体的关系。知识图谱已经在信息检索、用户关系挖掘等领域有了比较广泛的应用。但是现阶段还没有一个利用知识图谱解决答案摘要问题的方法。如果能通过知识图谱中的知识，解决 CQA 中答案摘要的问题，将是对答案摘要方法和知识图谱应用方法的一个促进。

答案摘要是解决 CQA 系统中答案不全面、水平不一致、高噪声等问题的一种有效手段。本课题提出了一种基于知识图谱解决答案摘要问题的方法，只借助于知识图谱中的实体对信息，不需要任何的有监督学习手段及语义相似度度量方式，对 CQA 系统中答案摘要问题有着比较大的意义。在理论上对摘要方法研究和知识图谱的应用方法有促进作用；在实际应用中对解决问答社区中答案不一致、不全面，增加问答对可重用性上有着重要的意义。

1.3 国内外研究现状

1.3.1 答案摘要国内外研究现状

现有的答案摘要研究主要有两种思路：沿用多文档摘要方法或基于查询的多文档摘要方法；基于某类特定问题选用特定的摘要方法和表现形式。

解决多文档摘要问题主要有基于图的方法，基于有监督学习方法和基于整数规划的摘要方法等。Textrank[3]和 Lexrank[4]是基于图的方法，可以看作是 Pagerank[5]的变种，不光考虑了实体权重，也将边作为考虑因素，缺点是不方便进行调试，更改参数对最终结果影响不大，对于长文摘要构建出图时间比较长，也比较费内存，也没有考虑到摘要的冗余问题。MMR[6]方法设定优化函数，尽可能的抽取出没有冗余、没有主题词或关键词重复的句子。Wang[7]基于 MMR 算法提出了一种自适应的 AMMR 算法，并在邮件摘要任务中进行实验，取得了良好的效果。

基于查询的多文档摘要(Query-Focused Multi-Document Summarization)与答案摘要在任务上比较相似，都是根据给定的问题，在限定字数下寻找与给定问题相关的摘要，不同之处在于基于查询的多文档摘要多是新闻语料，文章长度比 CQA 中的答案长、文档质量更高、文档数更多。Chali[8]提出了一种基于 SVM 的多模型融合方法，对 DUC06 的数据进行训练，将数据分成四份得到四个数据模型，对 DUC07 的数据进行预测，由于该方法对数据要求比较高，因而适用性较差。Li[9] 将图方法与主题模型相结合，对文档进行主题分析后使用基于图的方法进行句子排序，参数设置繁琐，重现困难。Ouyang[10] 则提出使用基于有监督的回归模型，学习答案句子与问题之间的相关程度，从而对句子进行排序

的方式进行摘要，由于需要进行大量的标注，使用范围窄，容易产生过拟合。McDonald[11] 提出使用基于整数规划的摘要方法，由于没有考虑到答案句子和问题之间的联系，虽然减少了冗余，但是选出来的句子容易和问题无关。

以上方法虽然不是应用在答案摘要领域，由于其方法的通用性，经过修改后仍然可以应用于答案摘要方法上。

答案摘要(Answer Summarization)最早由 Liu[12] 提出，通过判断问题类型，将问题分为了导航型(Navigational)、信息型(Informational)、事物型(Transactional)和社交型(Social)四类。导航型问题主要需求回答者提供某个网站的具体链接或者某个资源的 BT 资源；信息型主要询问方法、信息等，还包含了有固定答案的常识型问题和没有固定答案的动态问题；事物型主要询问观点，寻求意见，例如“买 Mac Pro 2015 款好不好？”这类问题;社交型为在 CQA 中有大量交友、聊天对话行为的问题。通过对问题类型的分析，Liu[12]总结出对于信息型问题是可以进行摘要的，并对其中的观点类问题和事实型问题使用不同的基于聚类的方法进行摘要。

对于观点型问题，有两种处理策略：直接抽取所有答案的整体观点，例如询问“谷歌和百度哪个更优秀？”，答案的内容有 100 个谷歌优秀、49 个百度优秀，则可以认为是谷歌更加优秀；另一种方法忽略观点型问题的类型特点，对观点型问题采用不区分类型的答案摘要方法。He[13] 则提出了一种解决 Yes、No 类型问题的方法，Yes、No 类型的问题可以看成是观点型问题，将问题和答案句子之间的相关性等进行特征映射，并转化成一个线性规划问题，通过特征映射来计算其相关度。Wang[14] 针对意见型问题，则将摘要问题转化成了一个排序问题，利用用户的 UGC 信息，如主题覆盖、作者回答主题的范围、答案内容范围等信息进行排序，抽取出所有答案的倾向情感，并与传统方法进行对比，取得了良好的效果。

Chan[15] 将 CQA 中的每个问题分解成了多个子问题，对每个子问题采用了有监督的学习方法，使用基于 L1 正则化的 CRF 方法，对每个答案中的句子进行学习，从而判别出选择哪个句子进行摘要。该方法对问题的选择限制比较大，因而通用性不好。Prande[16] 则更进一步，使用 SDPP(Structured Determinantal Point Processes)方法，对结果加以改进，选出不重复且可补全最佳答案的答案句子。

由于监督学习方法具有适用面窄，需要大量标注的缺点，近年来有部分学者提出使用无监督学习手段，例如整数规划的思想进行答案摘要方法研究。Liu[17]提出以二元连续词对(Bigram)和名词短语为摘要中的基本语义要素，结合

Yahoo answer 中已有的如作者权威性等信息，使用整数规划的方法，求出最大覆盖问题主题的句子集合，进行摘要。该方法具有速度较快、通用性好、易于重现的特点，但是没有考虑到问句和答案之间的联系，最终得到的摘要句子不能保证是与问题有关联的句子。在对于语义要素单位层面，Tomasoni[18]提出了一种基于基本语义要素(basic element，BE)的答案摘要方法，将词、词组、主题词等划分为基本语义要素，根据整数规划的方法得到摘要。由于语义要素的定义并不明确，与自然语言中的实际要素并没有良好的对应关系，因而说服力不大。

Natase[19]则对答案摘要中的语义要素有了较为明确的概念，把 Wikipedia[3]中的条目看作实体，将问题和答案表示实体的集合，通过扩散传播[20](Spreading Activation)的方法对实体的权重进行评定，根据实体的权重对句子的权重进行衡量，利用贪心的思想得到最优句子集合。该算法充分考虑了答案和问题之间的联系，通过答案和问题实体之间的关联程度，选择满足问题条件的句子，但是在最后选句子的过程中没有考虑到减弱句子冗余的问题。

基于查询的多文档摘要和 CQA 答案摘要都有选择的句子具有多余内容的问题，即句子中有一部分内容与问题不相关。因而句子压缩在答案摘要中也是很重要的一部分。在进行选择句子之前对句子进行压缩，去掉句子内对摘要结果没有帮助的部分。从而达到在字数尽可能少的情况下，尽可能多的包含于问题相关的内容。Qian[21] 提出使用基于图切割(Graph Cut)的方法对句子进行压缩，具有不需要训练、时间短、效果好的优点。Wang[22] 则对句子压缩问题进行了归纳和总结，分别实现了三种句子压缩方法：基于规则、基于句法树分析、基于序列分析的句子压缩方法。分别取得了比较好的效果，其中基于规则的句子压缩方法具有实现难度低、时间短、效果稳定、容易重现的特点，较为通用。

答案摘要的评测方法一般采用 ROUGE 值作为评价，ROUGE[23]是一种根据自动摘要和人工摘要之间共有的 N 元连续词对数目来评判摘要结果好坏的方法。ROUGE-N 利用连续 N 个非停顿词同现比例衡量摘要结果，ROUGE-SU-N 利用跳跃 N 个词同现来衡量摘要结果。

1.3.2 知识图谱及实体扩展国内外研究现状

近年来知识图谱及其相关领域的研究和应用越来越多。知识图谱的概念最早可追溯到以 Wordnet[24]为代表的语义网。Wordnet 以词为实体，研究词和词之间的关系，对语义相似度[25]、指代消解[26]、语义角色标注等研究都有重要的意

[3] http://www.wikipedia.org/

义。

随着互联网信息越来越多，单纯的词实体已经不能满足工业界和学术界的需求。以 Yago[27]、Dbpedia[28]和 Freebase[29]代表的基于人工、半人工、全自动的知识图谱应用越来越广泛，这些知识图谱根据 Wikipedia 中的内容进行抽取。利用知识图谱可以进行知识查询[27]、提高检索结果等工作。这些应用主要集中在利用知识图谱中实体与实体之间的关系，来提高计算实体相似度或者查询扩展等方面。与 Wordnet 一样，这些知识库具有严谨的分类体系，有着对实体的严格定义，在实际利用中仍然会带来一定的困难。

由于 Yago 等知识库具有严格的分类体系、对实体定义严格等特点，因此会大量的减弱知识库的表示能力。在 CQA 中有着大量的非严格定义的概念和实体，Yago 等类型的知识库在 CQA 系统中直接应用会有困难。以 ConceptNet[30-32]为代表的常识类知识库则能很好的解决此类问题，ConceptNet 对实体定义非常灵活、没有分类体系等优点，特别适合 CQA 这种具有大量噪声的文本。

Hsu[33]利用 ConceptNet 和 Wordnet 解决查询扩展问题(query expansion)。查询扩展多用于检索领域，通过增加和查询词相关的词或短语的方式提高检索效果。该方法利用扩散传播的方法，对扩展词的权重加以估计，得到可能的扩展词，该方法取得了良好的结果。Hsu[34]在上述方法的基础上加以改进，对扩展词加以标注，利用 SVM 对扩展词加以分类，得到了一个比较好的结果。Kotov[35]则将实体信息与相关性反馈进行融合，同时使用了有监督和无监督的方法进行对比，并提出了一个对实体边权重的衡量手段，取得了良好的效果。

由上述分析可知，利用知识图谱可以改进搜索结果，发现用户的查询意图，了解扩展后的查询词集合与被查询文档的相关性。因此将知识图谱做查询扩展的方法利用在答案摘要上，对问题进行扩展，衡量问题和答案句子之间的相关性。在答案摘要中由于没有查询，取而代之的是问题，因而称之为问句实体扩展。由于答案摘要是在固定文本上做摘要，而检索领域的文本则是开放文本，搜索范围更大，因而答案摘要更加适用于应用查询扩展的方法。

1.3.3 国内外研究现状小结

从上述介绍中可以看到，答案摘要方法研究出于初始阶段，没有较为通用的方法。多数方法都是直接采用或者改进多文档摘要、基于查询的多文档摘要的方法。因而没有充分的考虑到 CQA 文本的自身特点，也没有考虑答案和问题之间的相关性。片面的采取有监督学习方法，使得方法适用性不够广泛。

知识图谱具有先验知识，已经有知识图谱成功的应用在查询扩展上的先例。

经前文分析，知识图谱更加适合在答案摘要上查询扩展的相关工作，因而用知识图谱在答案摘要上进行查询扩展，衡量答案和问题之间的相关性，从而得到最优摘要的思路是可行的。

1.4 研究内容及章节安排

传统的基于有监督学习的答案摘要方法具有适应面窄、需要大量数据的缺点；基于图方法的摘要方法则没有考虑到答案句子和问题之间的联系。因而需要找到一种可以选出与问题相关且适应大规模应用的答案摘要方法。

本文将从知识库入手，力图寻找出一种可以不经过监督学习，建立问题与答案句子之间联系，从而得到答案摘要的方法。本文研究框架如图 1-1 所示。研究内容主要包括以下几个方面：

1. 研究基于知识图谱的问句实体扩展问题，将信息检索领域的查询扩展问题应用到答案摘要上，从问题中的实体出发，在知识图谱中找到跟问题实体相关的实体，与答案对应，从而建立答案与问题之间的联系。

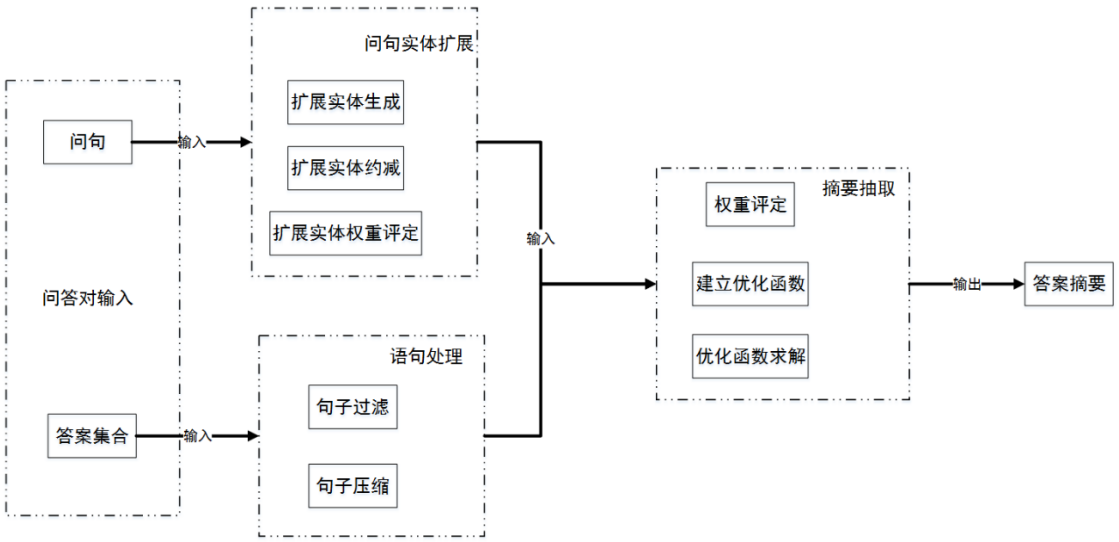

图 1-1 论文研究框架

2. 研究如何从根据问题和答案之间的联系，得到一个完备、与问题相关的摘要方法。将答案摘要问题转化成了整数规划问题，通过求解整数规划的最优解的方式，求解出最佳答案摘要句子。进一步地讨论了句子压缩、句子过滤、句子质量和未命中实体等方面对答案摘要效果的影响。

3. 将要完成一个答案摘要系统。在摘要系统中，将要有问题查询和问题摘要两个主要功能，并将摘要的结果以网页的形式进行展示。

章节安排如下：

第一章，绪论。指出了答案摘要的研究背景和意义。分别介绍了答案摘要方法、知识图谱和实体扩展方法的国内外研究现状，指出传统答案摘要方法的不足之处。表明现阶段还没有一个比较成熟的答案摘要方法。探究了知识图谱及实体扩展在答案摘要领域使用的可能性。

第二章，基于知识图谱的问句实体扩展方法。提出使用知识图谱来衡量问题和答案，并对问句进行实体扩展，从而得到问题和答案之间的相关性。使用 Pagerank 和启发式规则方法对扩展出的权重进行估计。

第三章，基于问句实体扩展和全局推理的摘要方法。研究如何从根据问题和答案之间的联系，得到一个完备、与问题相关的摘要方法。将答案摘要问题转化成了整数规划问题，通过求解整数规划的最优解的方式，求解出最佳答案摘要句子。进一步的讨论了句子压缩、句子过滤、句子质量和未命中实体等方面对答案摘要效果的影响。

第四章，答案摘要系统的设计与实现。实现了一个基于知识图谱的答案摘要系统，并添加了句子检索功能，充分的展示了基于知识图谱的答案摘要方法的效果。

第 2 章 基于知识图谱的问句实体扩展

2.1 引言

在答案摘要问题中，衡量问句和答案内容之间的关系是解决答案摘要问题的一个重要环节。传统的答案摘要方法[17]将问句和答案内容理解为由主题词、关键词、或者名词短语的集合。而这些手段都与现实中没有对应，因而只能依靠统计学习方法求解出主题词、关键词或者名词短语之间的联系，这样只能得到一个片面的知识集合。知识图谱中存在着大量的实体关系对，如果能将句子理解为实体集合，则可以利用知识图谱内的关系对求解出问句、答案内容之间的联系，从而判断问句和答案内容之间的相关程度。本章 2.2 节是基于实体标注的概念映射方法，对实体的概念加以定义，比较了与实体链接的不同，说明如何只借助词性标注方法从问句和答案中抽取出实体；2.3 节提出了一个基于知识图谱的问句实体扩展方法，说明如何找到与问句相关的实体；2.4 节提出了两种约减实体提高可信度的方法；2.5 节实验，给出了一种定量的分析扩展实体好坏的分析标准，并对基于 Pagerank 和启发式规则的实体约减方法加以评价；2.6 节总结本章之前的研究内容。

2.2 基于词性标注的概念映射方法

Wordnet[24]将实体理解为语法意义的物体，Yago[27]等将实体理解为现实世界中的出现的物体。本课题将实体的概念延伸，将实体定义为具有动作意义的动词和与其搭配的名词，如“buy food”，“drive to store”，“buy”等。这就使得表示日常世界的知识变得更加的合理和简单了。

由于实体的定义有所不同，因而从句子中抽取出实体的方法与传统的实体链接方法有所区别。传统的实体链接方法主要有生成候选实体、实体消歧两大步骤组成。在生成候选实体阶段，先对文本进行命名实体识别(Named Entity Recognition，NER)，找到人名、地名、机构名等名实体，然后在知识库中找出可能与该名实体相对应的候选实体。在实体消歧阶段，通过特定方法对候选实体进行排序，找到最后可能与名实体相对应的知识库中的实体。对于实体排序方法，主要有基于图方法、有监督学习方法两种。Hoffart[36]、Han[37]提出基于图的方法进行实体消歧，比较稳定；Bunescu[38]提出了一种基于 SVM 的有监督学习方式进行实体消歧，结果更高。AIDA[30]是一个基于 YAGO 的成型实体链

接系统，将普通文本中的实体链接到 YAGO 中。由于本文中的实体是概念级别的实体，主要识别动词、名词、动词短语、名词短语等，与传统实体链接方法只识别名实体不同，因而需要单独的实体链接方法，在本文中称为概念映射算法。

2.2.1 候选实体生成

CQA 文本具有长度短、随意性较大的特点，因而找到问题和答案中的实体变得更加困难。在传统方法中，找到候选实体通常通过命名实体识别(NER)来完成，但是命名实体识别在本问题中的效果并不理想。图 2-1 是 Yahoo answer 中 computer 一词的搜索结果[4]。可以看出，关于 computer 的问题中 computer 一词对问题的表义具有决定性的作用，因而识别出 computer 一词对解决摘要问题

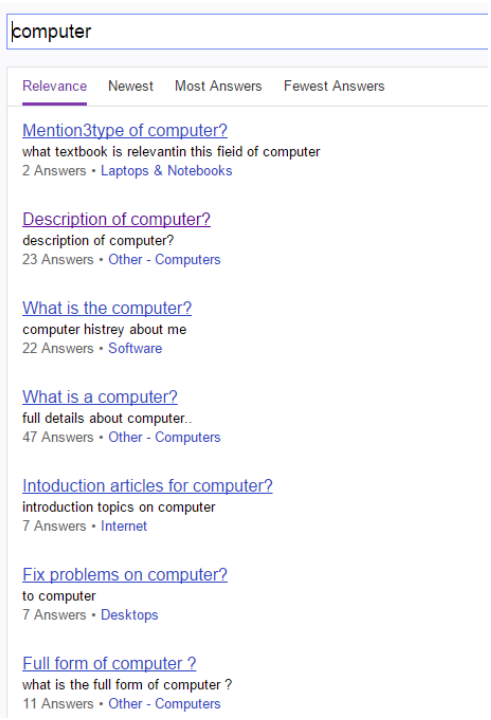

图 2-1 Yahoo answer 中 computer 的搜索结果

有决定性的作用，且 computer 可以很自然的被理解为一个实体。图 2-2 是 Stanford NER[5]对其中一个问题的命名实体识别的结果。可以看到，Stanford NER 并没有识别出任何的名实体，由于 Stanford NER 是比较著名的实体识别工具，

[4] https://answers.yahoo.com/search/search_result?fr=uh3_answers_vert_gs&type=2button&p=computer
[5] http://nlp.stanford.edu:8080/ner/process

因而可以认为 computer 很难被识别成一个名实体。这样就会使得问题和答案抽取的实体数变少，缺乏必要的语义要素，影响最终的结果。

为了尽可能多的抽取出有用的信息，本文采用只基于词性标注[39](Part of Speech Tagging, POS tagging)的手段，尽可能多的提取出名词、名词短语、动词、动词短语这些对问题和答案具有表义信息的词和短语。

表 2-1 词性转换规则

处理后的词性	原始的词性
N(名词)	NN, NNP, NNS
V(动词)	VB, VBD, VBG, VBN, VBP, VBZ
O(其他)	除上述以外的其他词性

在词性标注中，名词和动词具有多种不同的形式，如名词有 NNS、NN；动词有 VB、VBG、VBN 等形式，由于这些不同的词性均是同一单词在表义性上是没有大的区别的，例如词 eat、eating、ate 只是在词型上有区别，因而不妨将具有名词属性的词均识别成名词，将具有动词属性的词均识别成动词，由于其它类型的词在表义性上没有绝对的区别，因而均识别为其他(Other)，表 2-1 列出了对于词性的转换方式。

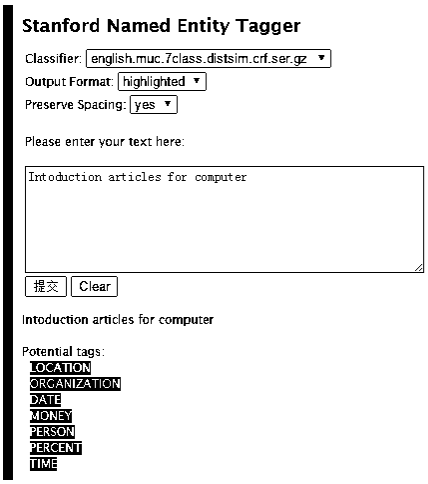

图 2-2 Stanford NER 对问题的识别结果

经过词性标注并进行词性转换后，所有的问题和答案句子均变成了只具有三种词性的词串。由于没有识别出名实体，句子中没有可以标示出哪些是实体的标识。在这里，认为名词、名词短语、动词、动词短语均是实体，可能成为

实体的 bigram 和 unigram 模式表 2-2 所示。

表 2-2 实体可能的模式

可能的模式
N
V
V + N
V + V
N + V
N + N

在这里，所有的名词和动词认为只有一个词组成，而名词短语和动词短语认定为由两个词组成，在实际中例如 “Southern Poverty Law Center” 这种单词也是一个名词短语，但是识别四元词对时间效率影响太大，所以大于二元的连续词不被认为是一个短语的。

如前文所述，这些实体在决定句子的表义性和与问题的相关性上可以起到决定性的作用。例如对于句子 “Today, computers can be made small enough to fit into a wrist watch and be powered from a watch battery.”，词性标注的结果如表 2-3 所示，括号内的为词性。我们可以看到具有表义性的词、短语为：computers、be made、be、made、fit、wrist、watch、wrist watch、be powered、powered、watch battery、watch、battery。

表 2-3 Stanford POS tagger 标注后的结果

句子
Today*(NN)*, computers*(NNS)* can*(MD)* be*(VB)* made*(VBN)* small*(JJ)* enough*(RB)* to fit*(VB)* into*(IN)* a*(DT)* wrist*(NN)* watch*(NN)* and*(CC)* be*(VB)* powered*(VBN)* from*(IN)* a*(DT)* watch*(NN)* battery*(NN)*

找到的候选实体可能含有包含关系，如 watch、battery、watch battery，三个，由于抽取出的短语可能不会在知识库中出现，因而对这三个词进行保留都是必要的。但是对于如 be 这种很明显的完全没有表义性的词，直接用停顿词表进行去除。

2.2.2 概念映射算法

知识库采用 ConceptNet5[32]。ConceptNet5 与 DBpedia、Yago2、Freebase 等知识库不同，ConceptNet5 是常识类知识库，以词、短语作为基本单位，每个词

或者短语都被认为是一个实体。DBpedia、Yago2 等知识库中对于 computer 的处理是将其当作了类别实体，求类别实体与普通实体相关程度的方法现阶段还比较缺少，不具有稳定性。而 ConceptNet5 则完全不具有结构性质，可以直接求出类别实体和普通实体间的相关程度。

出于对算法复杂度、时间效率和产生实体具有的噪声程度三方面考虑，本课题并没有采用先生成多个候选实体，然后消歧的传统办法。只验证实体在还原词干后，是否在知识库中作为是否是实体的判别依据。总体的实体链接算法如算法 2-1 所示。其中 1-4 步对应生成候选实体，第五步为实体链接。

算法 2-1 中，只对 bigram 以下的词进行了链接，对于 trigram 来说，生成候选实体并链接的时间复杂度为$O(N^3)$，在实际实验当中，所花费的时间远大于一分钟，超过了实际系统中所能负担的最大时间，因而没有采用。

算法 2-1 概念映射算法

输入：英语句子 sentence，停顿词表
输出：候选实体集合 entity_set
1 清理句子，去除掉句子中的特殊符号
2 对句子 sentence 进行序列化，得到序列 tokens
3 对 tokens 按照 2.2.1 节中的方法进行词性标注，得到(词，词性)为元组的序列 pos
4 根据 pos 产生二元词序列序列 bgm
5 FOR (词 1,词性 1) , (词 2,词性 2) IN bgm:
    5.1 对词 1 和词 2 分别进行词干还原，词 3=词 1+词 2
    5.2 如果词性 1 和词性 2 组成的元组为 N+N ,V+V,N+V 或 V+N 中的一个
        如果词 3 在知识库中，将词 3 加入 entity_set 中
        否则，如果词 1 或词 2 在知识库中，将其加入 entity_set
    5.3 如果词性 1 和词性 2 组成的元组为 N+O ,V+O,O+V 或 O+N 中的一个
        找到词性为名词或动词的词，如果在知识库中，将其加入 entity_set
    5.4 对于其他情况，不执行任何操作
6. 返回 entity_set

表 2-4 为问题和答案句子的实体链接结果，可以看到对于问句“How do motorcycles pollute？”来说，问题主要的内容集中在了 motorcycle 和 pollute 中，实体链接算法对于有用的实体成功的进行了识别。同时还可以发现，第一个答案句子中所识别出的实体有一定量的错误存在，这些错误会给后续工作带来一定的麻烦，一个可行的解决方式是利用停顿词表，对部分没有用的词进行过滤，另一种可行的方式是利用词在整体答案出现的词频进行过滤。表 2-4 可以说明

算法 2-1 成功的将问题中最重要的部分转化成了知识库中的实体。

2.3 基于实体关系的问句扩展算法

对于已经确定实体的问题句子和答案句子来说，如何找到问题句子和答案句子之间可能的关系是答案摘要中的核心问题。传统的答案摘要方法将求解问题句子和答案句子之间的关系问题看成了一个有监督问题。但在多领域、开放类别的情况下，对句子进行大量标注会产生训练不完整的情况，所需要标注的语句也会大量的增加。因而找到一个不需要对答案句子进行相关性标注的方法对于上线系统的成本控制和方法的泛化适应程度是至关重要的。

表 2-4 问题和答案句子实体链接举例

句子类型	句子内容	识别实体
问题句子	How do Motorcycles pollute?	do, motorcycle, pollute
答案句子	Susan Carpenter lays it all out in a los angeles times column	los_angele, susan,los, column，time, angeles, lay, carpenter
答案句子	Motorcycles and shooters are an appealing alternative to shelling out big bucks filling up the roof	Family, buck, scooter, motorcycle, shell,fill , sale，roof，go，alternative

本节将衡量问题和答案相关性的问题，转化为了衡量问题和答案中实体相关性的问题。通过判断答案中的实体与问题的相关度来间接推断答案中的句子与问题的相关度。

在 CQA 系统中，用户的问题主要是单句，或以多个短句子的形式出现，每个句子叙述的概念比较一致，所以将用户的问句看成一个单句，对单句进行实体链接。而答案句子每句之间叙述概念跨度比较大，不能这样做。

为了叙述方便，对一些概念进行形式化定义。假定问题句子为 SQ ，问题句子的实体集合为 $\mathrm{Q}=\{q_1,q_2,q_3,\cdots,q_n\}$ ，其中 q_i 为问题 Q 的第 i 个实体，问题实体共 n 个。

答案句子集合 $SA=\{sa_1,sa_2,sa_3,\cdots,sa_n\}$,其中 sa_i 为答案的第 i 个答案句子。答案句子实体集合 $A=\{a_1,a_2,\cdots,a_n\}$ ，表示每个答案句子中的实体，其中 a_i 为答案的某个句子，在这里为了讨论方便，忽略单个答案的概念，认为所有的答案句子都在一个答案内。则对每个答案句子 $a_i=\{e_{a_i1},e_{a_i2},\cdots,e_{a_iv_i}\}$,其中 e 为实体， v_i 为第 i 个句子的实体数量。答案中所有的实体表示为 EA。

实体扩展的思路如下，从问题 Q 出发，根据实体关系等因素的综合评定，在不参考答案实体集合 EA 的情况下，尽可能多的找到答案实体集合 EA 中的实体。例如对于问题“My computer won’t start?”[6]，在不看答案的情况下，也可以猜测出，最后摘要的句子可能会含有 bios，screen，cables，这些与电脑开机有关系的词汇，如表 2-5 所示，部分关键实体已经用黑线标、斜体标识出。

表 2-5 问题“my computer wont start”的答案

内容类型	内容
问题	My computer wont start?
答案句子	It could be a lot of things. You want to ***disconnect*** as much as you can and see if it will work then. Here is my suggestion: If your computer won’t even boot to the ***bios screen***, then the first thing that you should do is turn it off and pull the plug or, at least, turn off the ***power supply*** on the back of the computer. Let it sit for an hour or more and try again. If this cures the problem, then when you’re finished for the day and shut down, always turn off the power before you leave. Check to make sure all the ***cables***, inside and out, are seated properly and haven’t come loose. Don’t touch anything inside other than the cables. Remember to have the power off while you are working inside the computer. Unplug it from the wall. If these don’t work, then you probably have a bad piece of hardware. You try to isolate this …

问题的实体为：computer、start。如果从这两个实体可以找到上述关键实体，那么进而就可以认为含有上述关键实体的句子可能会是关键句子，更有可能成为答案摘要中的句子。

解决这种定位相关词语的方法比较多，如频繁项挖掘[40]、词对齐[41]等方法，但是无一例外，这些方法需要大量的标注语料进行训练，如果语料比较少或者有的问题类别没有训练到的话，对于未知类别做出的行为是不可预测的，不能充分满足要求

而知识图谱则可以充分的满足要求，知识图谱中含有大量实体和实体的关系对，如果我们可以知道哪些实体与 Q 有关系，我们就可以加以筛选，从而找到可能存在于答案中的实体。表 2-6 是在 ConceptNet5 中搜索 computer 得到的

[6] https://answers.yahoo.com/question/index?qid=20091017120601AAoEvsV

部分结果，带有下划线的实体是与表 2-5 中的问题相关的部分实体，如果可以从 computer 出发找到 cable、monitor、cpu、motherboard 等实体，那么我们就可以成功的获得问题与答案句子中的实体间的关系，从而找出问题与答案句子间的关系。

2.3.1 实体关系定义

通过分析句子中的实体，我们可以分析出句子所表述的内容。句子通过句子中的实体表述出句子的含义，通过句子中的实体与其他句子中的实体之间的关系表示句子和句子之间的关系，因此对实体和实体之间关系的定义有助于理解句子和句子之间的关系，进而找到与问句有关系的答案句子。

表 2-6 “Computer”在 ConceptNet5 中的搜索结果

实体 1	关系	实体 2
Computer	AtLocation	Office
Keyboard	PartOf	Computer
Computer	UsedFor	Play game
Motherboard	AtLocation	Computer
Monitor	PartOf	Computer
Cpu	PartOf	Computer
Ram	PartOf	Computer
Cable	PartOf	Computer

定义两个实体e_1、e_2在知识库中有关系 r 为$e_1 \overset{r}{\Leftrightarrow} e_2$。ConceptNet5 中共有 36 种关系，其中出现频率大于 2000 的共 26 种。这 26 种关系有表示同义的，如 Synonym, DefinedAs,DerivedFrom；有表示关联的，如: RelatedTo；有表示因果的，如: HasSubevent；有表示同位的，如: Antomy；有表示包含的，如: IsA,PartOf,HasA。如何更好的利用这些关系是关键问题。

实体可能有多种表达方式，例如 pollute 和 pollution，这两个实体在实际表达中可以互换，因而如果在问题实体中有 pollute 或 pollution 中任意一个，另一个也应该加上。故而定义同义关系：实体e_1、e_2满足同义关系即$e_1 \overset{Syn}{\Leftrightarrow} e_2$，则∃关系 r 使得 $e_1 \overset{r}{\Leftrightarrow} e_2$且 r 为 Synonym, DefinedAs, DerivedFrom 三种关系的一种。

为了求解一个实体可能关联的实体，定义关联关系：实体e_1、e_2满足关联

关系即$e_1 \overset{Rel}{\Leftrightarrow} e_2$，则∃关系 r 使得 $e_1 \overset{r}{\Leftrightarrow} e_2$。

2.3.2 问句实体扩展算法

实体扩展方法思路上如下：从 Q 出发，找到 Q 的同义实体扩展集合Q′，根据Q′找到所有与Q′相关联的实体 EQ。如果只根据 Q 找 Q 所有的关联实体，则由于Q本身有可能表述不全面，例如 pollute 实体的同义实体为 pollution 和 polluted。如果单纯只找 Q 的关联实体，则有可能存在实体与 pollution 相关联而与 pollute 不关联，但显然这样的实体也应该被扩展，因此必须要找到 Q 的同义实体，然后进行扩展。算法 2-2 所示。

算法 2-2 基于实体关系的实体扩展算法

输入：问题句子 SQ，同义层扩展层数 l1，关联层扩展层数 l2
输出：扩展后的问题实体集合 EQ
1. 根据算法 2-1 得到 SQ 的实体集合 Q
2. 当前实体 current = Q
3. 目标扩展集合 expand_entity = Q，层数 indx=0
4. while indx < l1
    4.1 对每个在 current 中的实体，找到所有跟该实体为同义关系的实体
    4.2 将这些实体加入到 expand_entity 中
    4.3 indx += 1
5. current = expand_entity，indx=0
6. while indx < l2
    6.1 对每个在 current 中的实体，找到所有跟该实体为关联关系的实体
    6.2 将这些实体加入到 expand_entity 中
    6.3 indx += 1
7. EQ = expand_entity
8.返回 EQ

在算法的第一步，对问题句子 SQ 进行实体链接，得到实体集合 Q。在第四步得到 Q 的同义实体扩展集合称为Q′。在第五步，得到Q′的关联关系的扩展集合 EQ。例如对于问题“How do Motorcycles pollute?”[7]，实体为 pollute、motorcycle。则 Q' 为 pollute, motorcycle, pollution, polluter, contamination, contaminative …等实体，经过同义层的扩充，可以得到 SQ 所能的表示的全部概念。而实体 car 和 motorcycle 和 pollute 都有着联系，可以认为 car 是答案摘要中可能出现的关键实体。表 2-7 展示了对于该问题最佳答案里 car 的出现频度，

[7] https://answers.yahoo.com/question/index?qid=20090409221945AAAIECK

可以看到，car 在最佳答案中出现了两次，从侧面说明 car 是答案摘要中的关键实体。

实体扩展总共进行了两次，第一次是从 Q 出发扩展出 Q 内实体所有为同义关系的实体，第二次从Q′出发，找到所有与其为关联关系的实体。将第一次的扩展成为同义层扩展，第二次的扩展称之为关联层扩展。

2.4 问句扩展实体约减算法

2.4.1 基于 Pagerank 的实体约减算法

通过算法 2-2 可以得到问题实体集合 Q 可能的答案空间 EQ,但是这个算法有两个缺点：1.只能找到哪个实体可能会成为答案中的实体，但是当 EQ 集合中的数量过大的时候，选出来的实体可信度就会降低，所以需要限制 EQ 集合的大小；2.对扩展出的实体没有重要性评定，后续利用比较困难。在算法 2-2 中，可以限制 EQ 集合的方法有两种，限制同义层扩展集合的大小和限制关联层扩展集合的大小。本节主要采取限制同义层扩展集合大小的方法。

表 2-7 “How do motorcycles pollute”答案中 car 与问题的联系

内容类型	内容
问题	How do Motorcycles pollute?
答案句子	Motorcycles and scooters are an appealing alternative to shelling out big bucks filling up the family truckster, which is one reason sales are going through the roof. But riding on two wheels may not be any more environmentally responsible than riding on four. Turns out the average motorcycle is 10 times more polluting per mile than a passenger ***car***, light truck or SUV. It seems counter-intuitive, because motorcycles are about twice as fuel-efficient as ***cars*** and emit a lot less C02. …

ConceptNet5 是自动抽取的知识库，里面实体关系会有一定的噪声，可能会存在着某些实体对应同一实体过多的情况。因而当同义实体集合Q′扩展过多的时候，如何对扩展的实体进行约减是一个关键的问题。

约减实体一个很常见的方法是实体赋予一定的权值，通过设定阈值或者最大个数的方法对实体进行约减。由于Q′可以看成是一个图，每个实体是一个结点，每条边是实体之间的关系，因而可以通过一些图方法来确定实体的权重。给实体赋予权值的另一个好处是在后续的摘要选句子中，实体的权重也可以作

为一个参考量。

算法 2-3 基于 Pagerank 的实体约减算法

输入：问题句子 SQ，同义层扩展层数 l1，关联层扩展层数 l2，
同义层最大实体个数 S1，知识图谱 K
输出：扩展后的问题实体集合 EQ
1. 根据算法 2-1 得到 SQ 的实体集合 Q
2. 当前实体 current = Q
3. 扩展实体 expand_entity = Q，层数 indx=0
4. while indx < l1
4.1 对每个在 current 中的实体，找到所有跟该实体为同义关系的实体
4.2 将这些实体加入到 expand_entity 中
4.3 indx += 1
5. 初始化图 G
6. 对任意两个在 expand_entity 中的实体对，查找其在不在知识图谱中，如果在加入到图 G 中
7 对图 G 进行 Pagerank，达到稳定后取权重前 S1 大的结点组成 SynQ，
expand_entity = SynQ
8. current= expand_entity，indx=0
9. while indx < l2
9.1 对每个在 current 中的实体，找到所有跟该实体为关联关系的实体
9.2 将这些实体加入到 expand_entity 中
9.3 indx += 1
10. EQ = expand_entity
11.返回 EQ

Pagerank[38]是衡量图中结点权重的一个比较常见的算法，因此经常被用做衡量网页的重要性，Google 使用得该算法。Pagerank 的内在思想很简单：越重要的结点的入度越多，结点的重要性由链接该节点的其他节点的数量和质量决定。公式(2-1)为更新公式

$$SW(p_i) = \frac{1-d}{N} + d \sum_{p_j \in M(p_i)} \frac{SW(p_j)}{L(p_j)} \tag{2-1}$$

其中p_i是图中结点，SW 是该节点的重要性。L 是该节点的出度函数，表示与该节点相邻且终点不是该节点的所有点的集合；M 是结点p_i所有入度的集合，表示除了该节点的入度集合内的点外，该节点的所有邻居。d 为 Pagerank 的参数，一般 0.85，在本实验汇中也取 0.85，1-d 表示从该节点跳出环的概率。

通过多次迭代，可以得到图的稳定状态，从而计算到图中每个节点的重要

程度。对于扩展集合Q′而言，得到经过约减的实体集合 SynQ，算法如算法 2-3 所示。

算法 2-4 基于启发式规则的实体约减算法

输入：问题句子 SQ，同义层扩展层数 l1，同义层最大实体个数 S1，关联层最大实体个数 S2，知识图谱 K

输出：扩展后的问题实体集合 EQ

1. 根据算法 2-1 得到 SQ 的实体集合 Q
2. 当前实体 current = Q
3. 目标扩展集合 expand_entity = Q ，层数 indx=0
4. while indx < l1
   4.1 对每个在 current 中的实体，找到所有跟该实体为同义关系的实体
   4.2 将这些实体加入到 expand_entity 中
   4.3 indx += 1
5. 初始化图 G
6. 对任意两个在 expand_entity 中的实体对，查找其在不在知识图谱中，如果在加入到图 G 中
7. 对图 G 进行 Pagerank，达到稳定后取权重前 S1 大的结点组成 SynQ，
   expand_entity = SynQ
8. 对任意在 SynQ 中的实体 entity，初始化空集合 RelQ，初始化哈希表 hash_syn
   8.1 找到所有与 entity 为关联关系的实体集合 rel_dict
      对 rel_dict 中任意实体 rel_entity，RelQ[rel_entity] = 0
      hash_syn[rel_entity] 添加序对(entity,E(entity,rel_entity))
   8.2 对 RelQ 中任意实体 rel_entity
      hash_syn[rel_entity] 表示当前实体与所有同义层实体的连接权重，根据公式(2-2)计算当前实体权重，得到结果为 w
      RelQ[rel_entity] = w
9. EQ = RelQ 中权重前 S2 大的实体集合
10. 返回 EQ

修改的部分主要集中在第六、七步中，将Q′构建成一个有向图，然后再 Pagerank，求出前 S1 大的结点。除了 Pagerank 外，还有一些如 HITS、连通度等方法也可以求出图的结点权重，但是由于稳定性和时间效率等原因不如 Pagerank 好。

2.4.2 基于启发式规则的实体约减算法

另一种可以约减实体的办法是在关联层对扩展的实体个数按照重要程度进

行约减。单纯套用 2.4.2 中的基于 Pagerank 的方法是不行的，因为在关联层扩展出来的实体一般都在几万个以上，Pagerank 所花费的时间非常长，因而不能采用。由于越与问题关联的实体，其与问题中的实体连接数越多，因而可以通过设定启发式函数的方法，对扩展的实体进行权重评定，然后对实体权重进行排序，选取前 K 大的实体。

为了快速计算关联层结点的权重，本课题采用了启发式算法。由于 2.3.2 中已经产生了同义层实体的权重，用同义层实体的权重估计关联层实体权重会节省计算时间，同时也更加合理。计算公式如下：

$$Weight_{e_i} = (count(R)+1)^2 \times \sum_{s_i \in SynQ} SW_{s_i} \times E(s_i, e_i) \tag{2-2}$$

其中 SynQ 表示经过同义层约减后的实体集合，count(R)表示实体e_i与 SynQ 中实体在知识库中有链接的个数，SW 表示实体经过 Pagerank 后的权重，E 是查找边权重的函数，如果s_i, e_i之间有链接则值是边的权重，没有则是 0。

公式 2-2 表示，对于关联层的扩展实体e_i，其与 SynQ 中有联系的实体数目越多，权重越高，其与 SynQ 中有联系的实体越重要，权重越高。这也符合我们的观察，对于问题中的关键实体，和它有关系的实体越可能与问题有关系。经过改动后的算法如算法 2-4 所示。

算法主要对第八步进行改进，哈希表的书写方式与 Python 中的字典类似。第八步与公式(2-2)中的计算顺序不一样，是先求出与Q′相关联的所有实体后，得到整个的关联层扩展实体，并记录关联层扩展实体与Q′相邻的集合，然后根据公式(2-2)计算出关联层实体的权重。这么做的原因是数据库中单次查询的开销比较大，通过查找同义层相邻的所有实体可以减少查询次数，增加执行速度。

2.5 实验结果及分析

2.5.1 数据集及评价指标介绍

2.5.1.1 评价指标介绍

实体扩展由于是中间结果，很难衡量其效果，因而采用一些比较简单的指标进行衡量。主要的衡量指标有：

1. 命中数：问题 Q 经过实体扩展后得到扩展集合 EQ，EQ 和答案中所有的实体集合 EA 相交所得集合的个数为命中数，即

$$hits = count(EA \cap EQ) \tag{2-3}$$

其中 count 函数为集合中元素的个数。

2. 覆盖率：EQ 与 EA 相交所得的集合的元素个数占 EA 内元素个数的百分比，即

$$coverage = \frac{count(EA \cap \mathrm{EQ})}{count(EA)} \tag{2-4}$$

对于多个问题的覆盖率，通过求它们的平均数来获得。

3. 扩展个数：即 EQ 中元素的个数。

4. 同义层剩余：经过同义层约减后，同义层扩展集合 SynQ 的元素个数。

其中命中数和覆盖率表征扩展实体效果，命中数和覆盖率越大，证明所扩展的实体越能覆盖到答案所叙述的句子。扩展个数和同义层剩余表征扩展实体的效率，扩展个数越少表明所扩展的实体覆盖率就越高，表明实体扩展效果越好。

表 2-8 语料集统计表 1

语料集	问题/平均答案个数	问题平均单词数	问题平均实体数
DUC	45/25	29	10
Yahoo answer	599/18	15	4

表 2-9 语料集统计表 2

语料集	答案平均单词数	答案平均实体数	答案平均句子数
DUC	537	137	21
Yahoo answer	107	21	5

2.5.1.2 数据集介绍

测试选用 DUC 2007 主题摘要公共数据集和 Yahoo answer 爬取的 599个问题-答案对两个数据集，分别简称为 DUC 和 Yahoo answer。

DUC 数据集包含 45 个主题，根据这些主题对文章进行摘要，任务与 CQA 相似,可以采用作为公共数据集。且 DUC 数据集与答案摘要的长问题比较相似，因此也具有参考价值。数据集 Yahoo answer 为开放问答社区，内有大量资源，语料差异性较大。

DUC 和 Yahoo answer 数据集的统计数据如表 2-8、2-9 所示，DUC 由于是新闻语料，所以句子数会长一些，Yahoo answer 由于答案质量有高有低，所以

单词数短，从而实体数相对少一些。

2.5.2 基于实体关系的问句扩展算法结果及分析

实体扩展结果如表 2-10、2-11 所示，Basic 为只用问题中的实体。T 和 R 分别表示同义层和关联层的扩展层数，后面跟随的数字表示层数。当关联层等于 2 的时候，计算时间比较长，没有采用。

表 2-10 Yahoo answer 数据集实体扩展实验

扩展策略	覆盖率	命中数	扩展个数
Basic-T0-R0	0.019	4	4
Basic-T1-R0	0.031	7	42
Basic-T2-R0	0.072	17	292
Basic-T1-R1	0.414	106	51499
Basic-T2-R1	0.667	175	210893
Basic-T0-R2	0.216	55	19748

通过观察表可以发现，两个数据集中的覆盖率比较相似，说明无论是对长文本还是短文本语料，方法的稳定性都比较好。在关联层等于零的情况下，扩展结果都不好，即小于百分之三，说明关联层起到了决定性的作用。对比两个数据集第六组和第四组实验结果可以发现，在有同义层结果的情况下，覆盖率的提升非常大，说明同义层扩展了问题实体 Q 的叙述范围。

表 2-11 DUC 数据集实体扩展实验

扩展策略	覆盖率	命中数	扩展个数
Basic-T0-R0	0.006	8	10
Basic-T1-R0	0.016	24	74
Basic-T2-R0	0.072	107	534
Basic-T1-R1	0.427	659	94088
Basic-T2-R1	0.718	1108	418339
Basic-T0-R2	0.182	274	25547

2.5.3 基于 Pagerank 的实体约减结果及分析

为了对比 Pagerank 结果，引入其他三种方法做对比实验，分别为：HITS，连通度(CC)和图内结点最小度数(KCore)，结果如表 2-12、2-13 所示。

Pagerank 和 HITS 内的数字表示同义层的最大实体个数，CC(10)表示只保留子图结点大于 10 的子图中的节点，KCore(4)表示只保留度大于 4 的结点。全部实验为同义层为一层，关联层为一层时所进行的优化。

表 2-12 Yahoo answer 数据集同义层优化结果

优化策略	覆盖率	命中数	扩展个数	同义层剩余
Basic-T1-R1	0.414	106	51499	32
Pagerank(40)	0.407	104	50434	32
Pagerank(30)	0.397	102	49496	27
Pagerank(20)	0.375	96	47918	19
HITS(30)	0.392	101	49465	27
CC(10)	0.407	104	51189	40
KCore(4)	0.363	92	48782	23

表 2-13 DUC 数据集同义层优化结果

优化策略	覆盖率	命中数	扩展个数	同义层剩余
Basic-T1-R1	0.427	659	94087	39
Pagerank(40)	0.403	624	90677	38
Pagerank(30)	0.390	603	88134	32
Pagerank(20)	0.362	558	84914	23
HITS(30)	0.384	593	88023	34
CC(10)	0.422	654	93946	71
KCore(4)	0.386	599	91045	43

由表可以看出，DUC 和 Yahoo answer 在覆盖率上是类似的，说明算法比较稳定。同时可以发现 Pagerank 在覆盖率上和其他的方法差不多，比连通度低是因为连通度的同义层剩余非常高。与 Basic-T1-R1 对比，扩展个数虽然有少量下降，但是不明显。所以可以表明利用 Pagerank 算法进行实体约减的稳定性更好，且由于 Pagerank 相较于连通度和最小度可以输出更为平滑的数值，因而在

具体应用中基于 Pagerank 的方法效果更好。

2.5.4 基于启发式规则的实体约减结果及分析

表 2-14、2-15 为基于启发式规则的实体约减方法观察在不同同义层最大个数和关联层实体最大个数时实体扩展结果。实验仍是在 Basic-T1-R1 的基础上进行实验。观察表可以发现，在相同同义层最大个数时，Yahoo answer 的结果会稍好一些，这主要是因为 DUC 数据集扩展出的实体比较多，留下相同个数时，DUC 数据集过滤掉同义层数目更大，因而 DUC 的结果会低一些。可以看到在扩展实体变为原来的十分之一时，覆盖率仍然保持着很高的数目，说明启发式规则所带来的方法的有效性，因而可以推断出公式 2-2 所得出的实体权重函数可以很好的评定实体权重。随着关联层最大实体个数下降，覆盖率下降的很快，在 4K 时覆盖率下滑幅度就比较大了。

表 2-14 Yahoo answer 数据集关联层不同策略优化结果

优化策略	覆盖率	命中数	扩展个数	同义层剩余
T(80)-R(10K)	0.373	95	6727	40
T(80)-R(8K)	0.365	93	5888	40
T(80)-R(6K)	0.351	90	4831	40
T(80)-R(4K)	0.321	82	3537	40
T(40)-R(10K)	0.368	94	6597	32
T(40)-R(8K)	0.361	92	5833	32
T(40)-R(6K)	0.347	89	4822	32
T(40)-R(4K)	0.318	81	3536	32
T(30)-R(10K)	0.360	92	6347	26
T(30)-R(8K)	0.354	90	5674	26
T(30)-R(6K)	0.343	87	4768	26
T(30)-R(4K)	0.314	80	3525	26

2.6 本章小结

本章针对 CQA 答案摘要的特点，提出了一种从实体的角度衡量建立问题和答案句子间联系的方法，将查询扩展的方法引入到答案摘要方法研究中。提出

了一种比较通用的 CQA 答案扩展方法，并根据实际需要，完成了一个适应数据集和知识库的实体链接方法。观察到问句实体扩展的效率和权重衡量问题，提出了基于图方法和更进一步基于启发式规则算法的扩展实体约减方法，效果好于基线方法，在扩展实体有非常大的数量减少时，仍能保证比较高的覆盖率。

表 2-15 DUC 数据集关联层不同策略优化结果

优化策略	覆盖率	命中数	扩展个数	同义层剩余
T(80)-R(10K)	0.330	504	9443	61
T(80)-R(8K)	0.318	484	7796	61
T(80)-R(6K)	0.299	455	5951	61
T(80)-R(4K)	0.270	409	4002	61
T(40)-R(10K)	0.317	484	9316	38
T(40)-R(8K)	0.305	464	7741	38
T(40)-R(6K)	0.288	439	5950	38
T(40)-R(4K)	0.261	395	4002	38
T(30)-R(10K)	0.308	470	9223	32
T(30)-R(8K)	0.297	451	7699	32
T(30)-R(6K)	0.279	426	5959	32
T(30)-R(4K)	0.253	385	4002	32

第 3 章 基于全局规划的答案摘要方法

3.1 引言

现有的答案摘要研究主要有两种思路：沿用多文档摘要方法或基于查询的多文档摘要方法；基于某类特定问题选用特定的摘要方法和表现形式。基于图方法的 Textrank、Lexrank 和基于整数规划的算法都没有考虑到答案和问句之间的联系，基于 SVM 的有监督学习算法或者基于 MMR 的算法则需依靠训练或计算语义相似度得出问句和答案之间的联系，因而答案摘要算法缺少一种可以同时兼顾答案和问句之间联系又无需训练的适用于所有问题类型的摘要算法。本章 3.2 节介绍以整数规划方法为代表的全局答案摘要算法的基本思想和优势，3.3 节介绍如何利用问句实体扩展后的信息来建立答案和问句之间的联系，确定优化函数利用整数规划的思想，从而得到既与问句有联系又无需人工标注的摘要。

知识图谱是基于自动构建的，因此不可避免的知识图谱中会存在噪声，例如边的权重不准，两个实体间有无关系标错等。对于只会利用少部分连接的任务来说，噪声不是一个很大的问题，Yago、Dbpedia 等知识图谱正确的比例都比较高。但是对于本课题来说，每个问题都可能会扩展出几万的实体，即使一个很小的比例权重不准，也可能会带来扩展的实体权重不准确的问题。本章 3.4 介绍如何利用句子质量和未命中实体两个特征来改善摘要的质量，从而优化摘要算法。3.5 节是实验，介绍了数据集和评价标准 ROUGE，得出了摘要算法的实际效果。3.6 节归纳了本章的基本内容。

3.2 基于全局规划的摘要方法思想

基于 Textrank、Lexrank 的传统摘要方法将答案句子看成一个节点，根据句子之间的关系组成网络，然后使用图方法对每个节点进行排序，得到每个节点的权重。传统摘要方法根据权重从高到低选择句子，这种方法带来的缺点是缺乏对摘要的整体考量，会选择出大量的重复句子。

近年来有部分学者提出使用全局规划的思想，例如整数规划的思想进行答案摘要方法研究。这种方法将句子看成由若干个基本单元组成，这些基本单元可以是句子中的词、短语、连续二元词对或者主题词等，设定基本单元在摘要中最大出现次数为优化函数。这个优化函数实质是一个整数规划函数，可以通

过固定的方法进行求解。利用整数规划的思想得到的摘要则不会存在摘要冗余的问题，因而近年来被大量学者所采用。全局规划方法的另一个好处是可以对同时对摘要中的句子和词进行限制，从而不仅保证摘要内句子不冗余，也能对句子与问题的切合度进行限定。

由于全局规划方法具有减少摘要冗余性、可以对句子选择增加限定增加限定等优点，本课题选用基于整数规划的摘要方法为基础框架，以答案中的问句扩展实体为基本单位，对答案摘要问题进行求解。以问句扩展实体为基本单位可以保证答案和摘要句子的关联性，选用基于整数规划的摘要框架则可以降低摘要的冗余性。

3.3 基于整数规划的答案摘要算法

摘要算法的主要工作集中在如何选取符合问句要求的句子。主要有对句子进行排序，用贪心的方法选句子和基于全局求出最大化目标函数选句子两种方法。由于需要选取符合题意的问句，因而基于贪心的方法可能存在着冗余度较高，答案点没有覆盖的情况。基于全局求出最大化目标函数的方法则先对问句基本要素进行抽取，然后建立优化函数，通过整数规划的求解最优句子集合。

本节先对句子进行句子压缩，得到质量更高的答案句子。通过实体扩展，找到答案中的命中实体，对部分信息量较少的句子加以过滤，并对命中实体的权重加以计算。然后确立优化函数，通过整数规划的方法求解出最佳句子。

3.3.1 基于问句实体扩展的摘要优化函数

在本课题中，答案句子由实体表示，有的实体贴近问题，有的实体不贴近问题，因此衡量实体重要性是判别答案句子和问题相关性的一个重要部分。本课题主要考虑实体的两方面因素：实体在经过扩展后的权重和实体本身在所有答案句子中的频度。

实体经过扩展后的权重主要是依据当前实体和同义层扩展实体 SynQ 的连接强度决定的，连接数越多或者权重越大，则当前实体的权重就越大，与问题的联系就越紧密。但是该实体与答案句子之间的联系则不能保证。

实体在所有答案句子中出现的频度考虑了实体出现次数越多，说明实体在越关联整个答案，实体的权重就应该越大。但是该实体和问题的相关性则不能保证。

由于上述两种因素具有互相促进的作用，因此将上述两种因素联合起来构成一个联合的权重对摘要的最终结果会有提高。

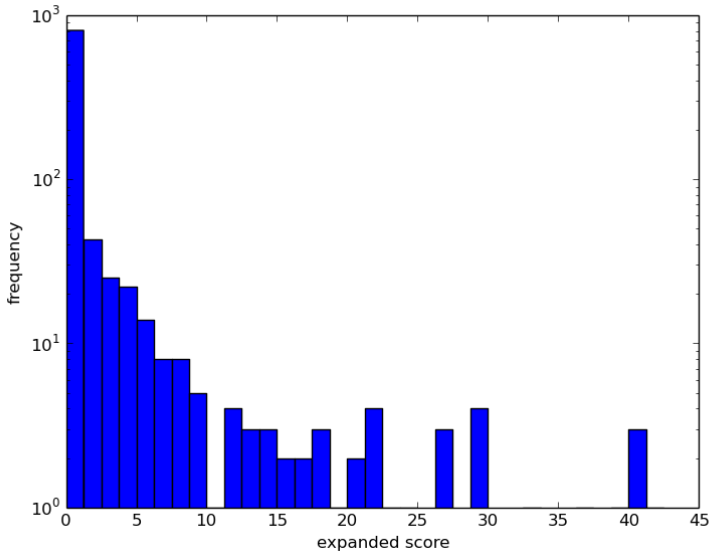

图 3-1 关联层实体分数直方图

图 3-1、3-2 分别为 DUC07 数据集中 D701 文章的关联层命中实体分数和答案实体频率的直方图，由图中可以发现，实体分数和实体频率均为长尾数据，对于实体分数来说，分布主要为 0-5000；实体频率主要分布为 0-120。实体分数与频率的范围不统一，将这两种特征融合到一起比较困难。

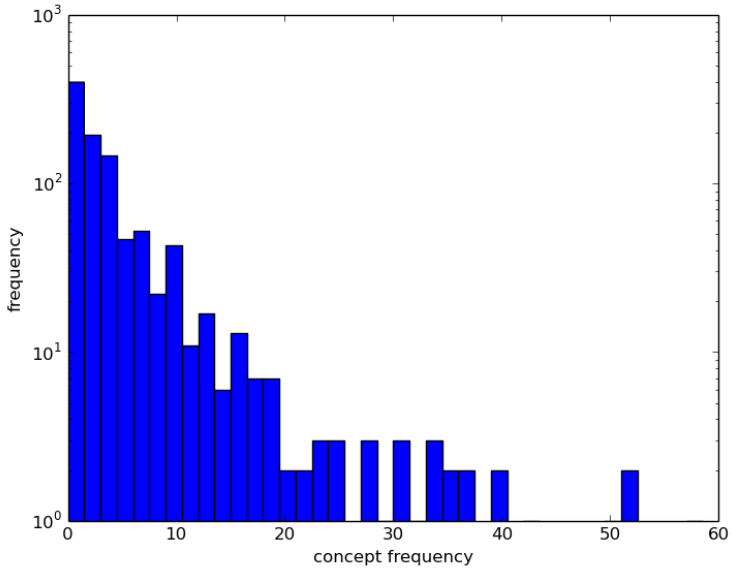

图 3-2 答案实体频率直方图

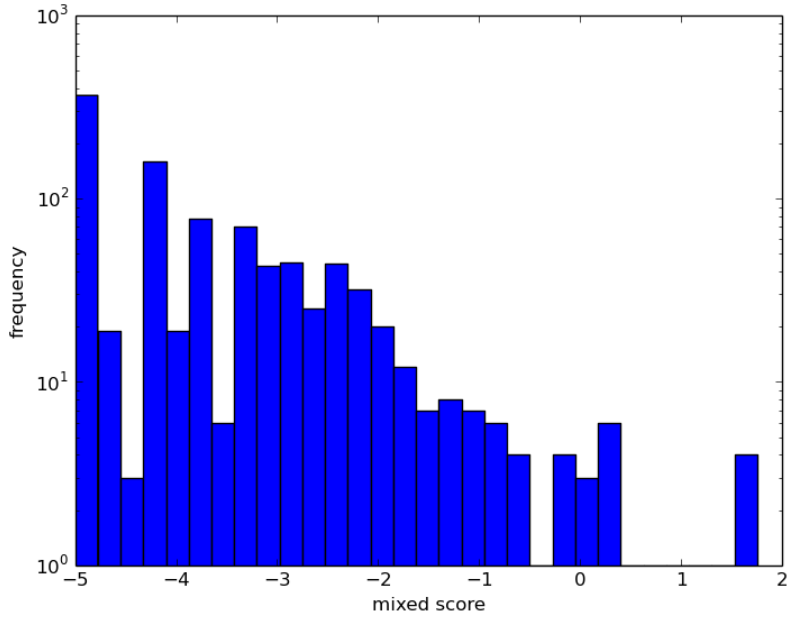

图 3-3 放缩后的实体权重

对于实体分数来说，直接进行成比例放缩会带来问题。小权重的实体在经过大比例放缩后，会接近为 0。由于这些小权重的实体占有的比重较大，因此对小权重有区分度会提升摘要的结果。因而不能采用直接放缩的方式。一种比较常用的放缩方式是取 sigmoid，将原来的权重缩小为 0-1 之间的实数，这同样会带来问题。对于大权重的实体，例如 300 以上的实体，sigmoid 之后得分全部为 1，使得 300 以上得分的实体失去判别作用。而这些大权重的实体对于关联答案有决定性的作用，所以 sigmoid 这种放缩方式不适合本问题。本课题选取了一种不太常见的放缩手段，借鉴于 TF-IDF 的思想，对实体权重取对数，进行放缩。在实体权重过小的时候，取对数后权重会变得特别小，因而对实体权重整体又增加了一个常数，因而实体扩展后的权重经过放缩变为：

$$expansion\_weight = \log(ow + \beta) \tag{3-1}$$

其中，ow 为实体扩展后的权重，β 为常量。

由于 CQA 语料句子数目差别比较大，直接用实体出现频率作为权重不能衡量出实体在整个答案中的重要程度。采用实体频率在所有句子中所占的比例衡量更加恰当，并且同样的采用了取对数的归一化手段。因而实体频率的重要性变成了：

$$freq\_weight = \log(\frac{freq}{D}) \tag{3-2}$$

其中 freq 为实体出现的频率，D 为答案句子总数。

因此实体权重公式变为：

$$w = \log(ow + \beta) + \alpha \cdot \log(\frac{freq}{D}) \tag{3-3}$$

其中α为加权系数，用来调节两个权重的平衡程度。放缩后的实体权重如图 3-3 所示。

在句子表示成一系列带有权重的实体后，答案摘要的目的就是尽可能的覆盖可能是答案的实体，使得高质量的实体和句子尽可能进入到摘要中。也就是说，要找到一个句子子集，满足以下两个条件：长度限制，小于规定字数；摘要权重最大化，用以保证生成摘要的多样性、覆盖面以及生成摘要跟问句的相关程度。

因此答案摘要问题实际上就可以看成一个带背包条件的最大覆盖问题(MCKP)[42]。MCKP 问题是 NP-hard 问题，没有最优解，一般是通过线性规划方法得到最优解，然后在最优解附近寻找整数解的方法进行解决。因此优化函数设定为：

$$Objective: \quad \max \sum_i w_i \cdot x_i \tag{3-4}$$

$$s.t. \quad \sum_j l_j \cdot y_j \le L \tag{3-5}$$

$$\sum_j OCC_{ij} \cdot y_j = x_i \quad \forall i \tag{3-6}$$

$$w_i = \log(ow_i + \beta) + \alpha \cdot \log(\frac{freq}{D}) \tag{3-7}$$

$$x_i \in N, y_i, OCC_{ij} \in \{0,1\} \quad \forall i, j \tag{3-8}$$

其中s_j是答案句子集合 A 内第 j 个句子，其具有长度为l_j。e_i是命中实体中的第 i 个实体，其具有权重为w_i。这里对于句子和实体的标号是随机标的，没有特定含义。假设生成摘要为 S，x_i,y_j分别表明实体e_i和句子s_j是否会出现在 S 中。x 是自然数，当为 0 时，表示实体e_i不会出现在 S 中，其他情况表示在 S 中的出现次数。y_j为 0 时表明句子s_j不会出现在 S 中，否则表明句子在 S 中出现了一次，

显然句子s_j是不会出现两次的。OCC 是二维矩阵，OCC_{ij}表明实体e_i是否在句子s_j中，是则为 1，不是为 0。

目标函数的目的是求出尽可能多的积累权重。与传统基于整数规划侧重于更大范围的覆盖主题词不同的是，目标函数对于重复出现的主题也进行叠加权重。

3.3.2 句子压缩和句子过滤

3.3.3.1 句子压缩

句子压缩(sentence compression)也是解决摘要问题的一种处理方法，在进行选择句子之前对句子进行压缩，去掉句子内对摘要结果没有帮助的部分。从而达到在字数尽可能少的情况下，尽可能多的包含于问题相关的内容的目的。句子压缩主流方法有基于规则[43]、基于句法树[44]、基于学习策略[22]三种。鉴于实现难度，本课题选用基于规则的句子压缩方法，压缩规则见表 3-1。

表 3-1 句子压缩规则

规则编号	规则	例子
1	去掉新闻类语料的开头	[MOSCOW,October 19(Xinhua)--]Russian fedral troops …
2	去掉相对日期	I went out [on Tuesday]
3	去掉句子内的修饰语	“I liking skiing”, [she said]
4	去掉句子开头的形容词、副词修饰语	[Interesting], he will go tomorrow.
5	去掉非限制性定语从句	He, [who was the president of united states], died lasted
6	去掉句子前的动名词短语	[Starting in 1990], there is only one big country.
7	去掉括号内的内容	Baidu[(www.baidu.com)] is a good company.

经过实验，规则 1、6、7 对摘要结果有改进最用，其余的没有显著效果。

3.3.3.2 句子过滤

句子过滤的目的是过滤掉一部分明显不会被选为摘要的句子。特别长的句子如果不是摘要句子，则会有比较多的单词没有命中，对最终的结果影响较大。对于特别短的句子，例如只出现了一个单词的句子，可能确实与问题十分相关，但是本身表义性比较差，对结果也会有降低的作用。

本课题选用句子内实体数目和句子长度两个比较容易衡量的指标来过滤句

子于长问题的摘要，过滤掉实体数小于 6 或句子长度小于 8 或句子长度大于 50 的句子；对于短问题的摘要，过滤实体数小于 2 或句子长度小于 5 或句子长度大于 20 的句子。经过实验证明，过滤句子对最终的摘要效果有提升作用。

3.4.3 摘要抽取算法

摘要生成算法如算法 3-1 所示。

算法 3-1 知识图谱答案摘要算法(Knowledge based summarization, KS)

输入：问题 SQ，答案句子列表 A
输出：摘要 S
1 对 A 中所有的句子进行句子压缩，得到清理过的句子集合 CA
2 对问题 SQ 进行实体链接，得到问题实体列表 Q
3 对 CA 中所有的句子进行实体链接，得到句子实体列表 EA 和答案的全部实体集合 TA
4 对 Q 进行实体链接，得到扩展实体集合 EQ
5 进行句子过滤，过滤掉不符合过滤条件的句子，得到新的句子实体列表 NEA 和新的答案的全部实体集合 NTA，并对 NEA 中的句子进行标号
6 对 TA 和 EQ 取交集，得到命中实体 HA，并对命中实体进行标号
7 构建句子-实体出现的矩阵 OCC
8 根据公式 3-4 到公式 3-8 的优化函数,对 NEA 和 OCC 进行求解出最佳句子集合 S
9 返回摘要 S

算法说明：在 1-4 步进行预处理、实体链接和问句实体扩展，在 5-7 步进行构建摘要优化函数的准备工作，第 8 步求解目标函数最优的句子集合，即为摘要。将该算法定义为 KS 算法(Knowledge based summarization)。

命中实体保证了答案句子中的实体可以和问句有联系，优化函数保证了摘要会尽可能多的覆盖关键实体，句子压缩和句子过滤则可以去除答案中低质量的句子，因而可以得到一个好的摘要算法。

3.4 答案摘要算法优化

由于 CQA 答案质量不一致的问题比较严重，每个句子质量差异比较大，算法 2-1 也没有考虑对高质量的句子进行一个加分。

在算法 3-1 的第 6 步中,对 TA 和 EQ 进行取交集,得到了命中实体的列表，由表 2-9、2-10 所示，命中实体覆盖率在 40%左右，对于未命中的 60%实体来说，有的是与问题完全无关的实体，有的和问题相关。因而对未命中的实体，如果仍能赋予其以一定权重，则可能会对摘要总体结果有所提升。

综上，对句子质量和未命中实体如果能够有一定的策略加以利用，摘要的最终结果可能会大大提高。

3.4.1 基于句子质量评价的答案摘要优化算法

在 3.3.2 节中对低质量的句子进行过滤或者压缩，由于其中所使用的句子过滤和压缩方法仍然会有低质量的句子或者无关内容没有去掉的情况，因而仍会出现一定比例的低质量句子。对于这些没有被过滤掉的低质量的句子，对高质量的句子增加权重可以提高不同质量的句子之间的差别，从而能使其有更好的区分。本课题将答案句子表示成实体，认为答案句子质量主要和两方面因素相关：答案句子的实体和句子内实体的权重。

在单位句子长度下，阐述与问题相关实体数目更多的句子质量更高。例如对于问题“电脑坏了怎么办”，同时带有主板、显卡的句子显然比只带有显卡或者主板的句子质量高、覆盖广，这样的句子与问题显然更相关。因而答案句子实体的数目对句子质量有一个正加成。

对于问句“西红柿和什么搭配比较好”来说，虽然鸡蛋和锅这两个实体和西红柿都比较相关，但显然鸡蛋同时和西红柿、搭配这两个实体的关联程度都比锅这个实体强。因此可以推断出，如果答案句子中的实体权重比较大，那么答案句子的质量也比较高。因此答案句子内实体的权重总和越大，答案句子的质量越高。

同样可能增高答案句子质量的还有句子长度，长句子对问题的回答可能更加全面，但同时会使实体数目/句子长度减少，会减弱句子的质量。在实际实验中，句子的长度跟最终摘要的结果无关。

据此，句子质量的评定函数为：

$$t_j = b_1 \cdot count(a_j) + b_2 \cdot \sum_{c \in a_j} w_c \tag{3-9}$$

其中t_j为句子s_j的质量，a_j为句子s_j的实体列表，count 为输出集合内元素个数的函数，w_c表示实体 c 的权重。两个 b 是加权系数，在实践中$b_1 = 2, b_2 = 0.5$时，效果最好。

因此摘要的优化函数变为了：

$$Objective: \quad \max \sum_i w_i \cdot x_i + t_j \cdot y_j \tag{3-10}$$

$$s.t. \quad \sum_j l_j \cdot y_j \le L \tag{3-11}$$

$$\sum_j OCC_{ij} \cdot y_j = x_i \quad \forall i \tag{3-12}$$

$$w_i = \log(ow_i + \beta) + \alpha \cdot \log(\frac{freq}{D}) \tag{3-13}$$

$$t_j = b_1 \cdot count(a_j) + b_2 \cdot \sum_{c \in a_j} w_c \tag{3-14}$$

$$x_i \in N, y_i, OCC_{ij} \in \{0,1\} \quad \forall i, j \tag{3-15}$$

3.4.2 基于未命中实体的答案摘要优化算法

由表 2-9、2-10 可以看到，经过扩展实体有 40%左右的命中率，这说明有 60%的实体没有命中。由于知识库中有噪声存在，在这 60%的实体中，应该还有少量的实体也是跟问题相关的。因而对于这些未命中实体也能有所筛选，并对筛选后的实体权重也能有所衡量，会对答案摘要结果有所提高。

由于未命中实体本身和问题建立不了联系，因而无法从问题的角度出发对这些答案实体进行衡量，所以只能从答案的角度考虑什么实体可能对最终结果有帮助，什么实体没有帮助。由 3.2.1 节可以知道，高频词对答案摘要的影响是正相关的，低频词有可能与问题无关，有可能是噪声。因而一个比较简单直接的方式是设定一个频率阈值，对于大于这个频率阈值的实体则保留，对于低于这个频率的实体则过滤掉。在本课题中，对于长文本设定的阈值是 4，对于短文本设定的阈值是 5。

如何衡量这些被选中的未命中实体权重也是一个关键问题，因为这些实体和频度有关，因而不妨假设他们的扩展权重也和频率相关，因而假设其权重为：

$$w = \log(freq + \beta) + \log(\frac{freq}{N}) \tag{3-16}$$

其中 freq 为实体频率，其实就是将扩展实体的权重替换为了实体的频率。因此优化函数变为了：

$$Objective: \quad \max \sum_i w_i \cdot x_i + t_j \cdot y_j \tag{3-17}$$

$$s.t. \quad \sum_j l_j \cdot y_j \le L \tag{3-18}$$

$$\sum_j OCC_{ij} \cdot y_j = x_i \quad \forall i \tag{3-19}$$

$$w_i = \log(ww_i + \beta) + \alpha \cdot \log(\frac{freq}{D}) \tag{3-20}$$

$$ww_i = ow_i \quad if \quad w_i \in hit\_entity \quad else \quad freq \tag{3-21}$$

$$t_j = b_1 \cdot count(a_j) + b_2 \cdot \sum_{c \in a_j} w_c \tag{3-22}$$

$$x_i \in N, y_i, OCC_{ij} \in \{0,1\} \quad \forall i, j \tag{3-23}$$

其中 hit_entity 对应算法 3-1 中的 HA，即命中实体。优化函数同时考虑了句子质量和未命中实体对摘要结果的影响。增加了句子质量和未命中实体权重衡量的算法如算法 3-2 所示。该算法为 KSSU(Knowledge based summarization with sentence quality and unseen entity)。

算法 3-2 改进的知识图谱答案摘要算法(KSSU)

输入：问题 SQ，答案句子列表 A

输出：摘要 S

1 对 A 中所有的句子进行句子压缩，得到清理过的句子集合 CA
2 对问题 SQ 进行实体链接，得到问题实体列表 Q
3 对 CA 中所有的句子进行实体链接，得到句子实体列表 EA 和答案的全部实体集合 TA
4 对 Q 进行实体链接，得到扩展实体集合 EQ
5 进行句子过滤，过滤掉不符合过滤条件的句子，得到新的句子实体列表 NEA 和新的答案的全部实体集合 NTA，并对 NEA 中的句子进行标号
6 对 TA 和 EQ 取交集，得到命中实体 HA，并对命中实体进行标号
7 对新的答案句子集合 NEA，根据公式 3-9 计算出句子质量 AQ
8 对未命中实体，即在 TA 中但不在 EQ 中的实体集合 UE，进行未命中实体过滤，并计算其权重，将其加入到命中实体集合中
9 构建句子-实体出现的矩阵 OCC
10 根据公式 3-4 到公式 3-8 的优化函数，对 AQ、NEA 和 OCC 进行求解出最佳句子集合 S
11 返回摘要 S

算法增加了对句子质量的计算。在第八步，对未命中实体仍然计算权重，将其加入到命中实体集合 HA 中。

3.5 实验结果及分析

3.5.1 数据集与评价指标

3.5.1.1 数据集介绍

数据集有两组，分别为 DUC 2007 摘要数据集和本课题标注的 Yahoo answer 40 个问题答案对的语料。其中 DUC07 语料为长文本，评测标准较为全面，对应着答案长度较长时的情况。Yahoo answer 40 问题对的语料与正常 CQA 问答对长度一致。DUC07 每篇文章对应四篇摘要结果，进行综合统计。Yahoo answer 由于是短文本，每个问答对对应一篇摘要结果。DUC07 和 Yahoo answer 的文本信息统计表见表 2-8、2-9。

3.5.1.2 评价指标介绍

评价指标选用多文档摘要比较通用的 ROUGE[9]作为评测标准，ROUGE-N 利用连续 N 个非停顿词同现比例衡量摘要结果，ROUGE-SU-N 利用跳跃 N 个词同现来衡量摘要结果。ROUGE-N 的公式如下：

$$\text{ROUGE-N} = \frac{\sum_{S \in ReferemceSummaries} \sum_{gram_n \in S} Count_{match}(gram_n)}{\sum_{S \in ReferemceSummaries} \sum_{gram_n \in S} Count(gram_n)} \tag{3-24}$$

Count 为计数函数，$gram_n$为连续 n 个词同现。

本实验主要采用 ROUGE-1、ROUGE-2、ROUGE-SU4 三个指标来衡量摘要最终的结果，每个指标测试其准确率、召回率、F 值。置信度为 95%。ROUGE 脚本执行命令为：“ROUGE-1.5.5.pl –n 4 –w 1.2 –m -2 4 –u –c 95 –r 1000 –f A –p 0.5 –t 0 –a -d”

3.5.2 实验结果及分析

对于 DUC07 语料，根据该评测要求，摘要字数选择 250 词；对 Yahoo answer 语料，由于每个问题-答案对的字数差异比较大，有的问题需求比较多的字数，有的答案则只需要几个字，因此选择所有答案总字数的三分之一，如果超过 150 词则定为 150 词，作为摘要字数限制。

3.5.2.1 与基线方法对比

表 3-2、3-3 为 KSSU 算法与基线方法的对比，基线方法选择摘要方法中比较常见的 Textrank 和 Lexrank 算法两种，已经全部转化为百分比的形式。Yahoo answer 数据集另外附加了一组最佳答案，最佳答案为 Yahoo answer 网站中经过

题主或群众投票被认为最符合题目要求的答案。

表 3-2 DUC07 摘要结果

评价指标	Textrank	Lexrank	KSSU
ROUGE1_P	34.171	34.15	**41.083**
ROUGE1_R	41.993	43.36	**43.183**
ROUGE1_F	37.644	38.11	**42.093**
ROUGE2_P	6.707	7.325	**10.883**
ROUGE2_R	8.251	9.280	**11.429**
ROUGE2_F	7.392	8.164	**11.145**
ROUGE-SU4_P	11.823	12.610	**15.852**
ROUGE-SU4_R	14.566	16.038	**16.661**
ROUGE-SU4_F	13.037	14.081	**16.240**

表 3-3 Yahoo answer 摘要结果

评价指标	Textrank	Lexrank	最佳答案	KSSU
ROUGE1_P	47.721	**50.355**	48.534	50.286
ROUGE1_R	41.013	40.966	37.165	**67.148**
ROUGE1_F	42.456	43.441	36.906	**56.244**
ROUGE2_P	27.171	28.211	28.417	**34.855**
ROUGE2_R	23.165	22.900	21.761	**48.264**
ROUGE2_F	23.867	24.224	21.838	**39.614**
ROUGE-SU4_P	29.361	30.726	30.320	**34.800**
ROUGE-SU4_R	24.865	24.856	23.299	**48.069**
ROUGE-SU4_F	25.662	26.295	23.157	**39.468**

DUC07 和 Yahoo answer 两个数据集整体上的值差的比较多的原因是句子数量不同，因而选择难度不同。由表中可以看到，Textrank 和 Lexrank 整体效果比最佳答案稍好一些，但是仍然和最佳答案差距不大。其中可以看出 Textrank 结果逊于 Lexrank。

对比基线方法，KSSU 算法在 Yahoo answer 数据集上超出基线方法和最佳答案非常多，其中 ROUGE1 的 F 值超出了其他方法将近 15 个百分点，说明 KSSU

算法能准确的找到与问题相关的实体并进行定位，可以找到近六成最佳摘要中会出现的单词。对比 ROUGE2 的结果可以发现 ROUGE2 的 F 值相较与其他方法也同样高出了 15 个百分点，在实验过程中，发现 ROUGE2 的提升相对于 ROUGE1 来说更加困难，因此说明 KSSU 找出的句子非常符合最佳答案摘要中的句子。说明对于短文本，KSSU 方法是完全可行的。

DUC07 数据集上的结果超出基线方法比较多，在各项指标上提升都比较大，完全没有出现因为文档长度的改变而变得效果变差的状况。因此，KSSU 算法是适合于长答案的。

3.5.2.2 实体权重参数选取

对于公式 3-3，由于α、β直接确定实体的权重，因而选取α、β的值对最后的实验结果影响非常大。对于这两个变量来说，可以采用网格搜索的办法进行选取。但 KSSU 算法对 DUC07 45 主题进行摘要需要 20 分钟的时间，如果各去 20 组，则需要花费 8000 分钟，大大超出了实验可能承受的范围。而且实验中采取了 Pagerank 方法，每次迭代出的实体权值都稍有差别，且 MCKP 还是 NP-hard 问题，不可能每次都有最优解，因此每次实验的结果都会有轻微的差异，这对网格搜索来说是致命的。

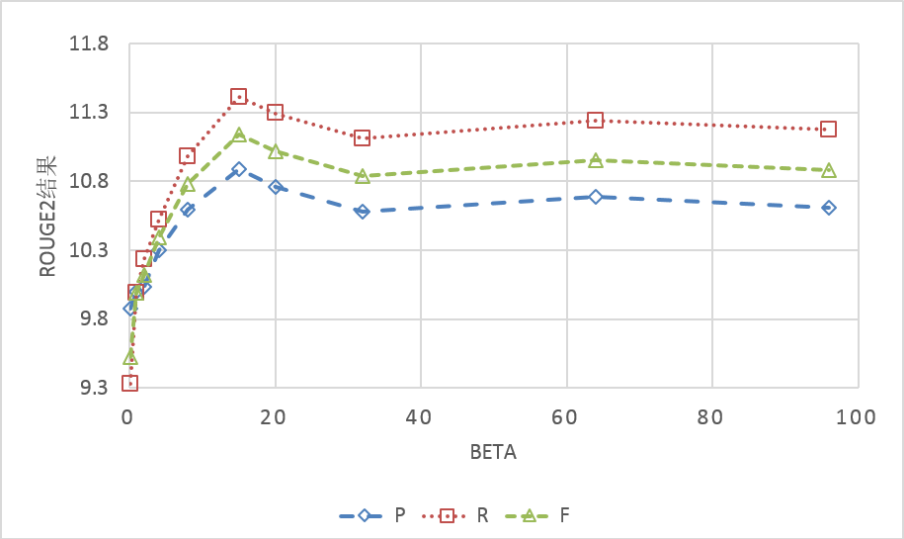

图 3-4 β对摘要效果的影响

一个比较简单的解决方式是直接采取控制变量的方式，固定一个变量，对另一个变量求得最优解。

图 3-4、3-5 分别展示了β、α对摘要结果的影响，其中 3-4 固定α为 1.1 画出

了 DUC07 语料的 ROUGE2 三个值得结果，可以比较明显的得出β = 15时结果是最优的。同样的，可以发现对于图 3-5 中α = 1.1时结果最优。

3.5.2.3 各项特征对比

本节主要测试基本方法与增加了句子质量特征、未命中实体特征这两种特征的对比。表 3-4、3-5 为不同特征叠加的结果，其中 KS 为不加句子质量和未出现实体两个特征的基本方法，KS+SQ 为增加了句子质量方法，KS+UE 为增加了未出现实体权重的方法。

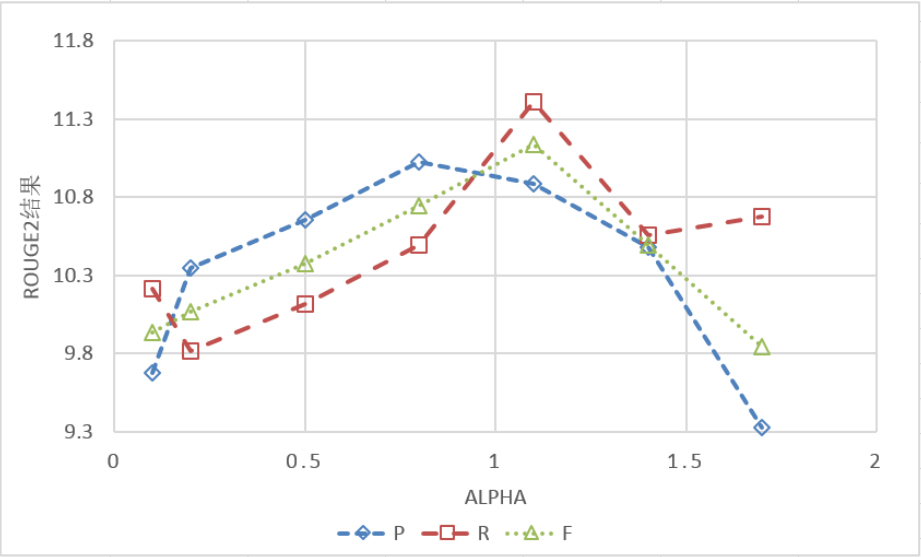

图 3-5 α 对摘要结果的影响

表 3-4 DUC07 不同特征叠加结果

评价指标	KS	KS+SQ	KS+UE	KSSU
ROUGE1_P	**42.566**	40.867	41.079	41.083
ROUGE1_R	39.459	42.193	40.300	**43.183**
ROUGE1_F	40.630	41.481	40.901	**42.093**
ROUGE2_P	10.438	10.237	10.290	**10.883**
ROUGE2_R	9.847	10.606	10.532	**11.429**
ROUGE2_F	10.063	10.413	10.327	**11.145**
ROUGE-SU4_P	15.817	15.450	15.432	**15.852**
ROUGE-SU4_R	14.792	15.975	15.030	**16.661**
ROUGE-SU4_F	15.172	15.695	15.224	**16.240**

可以看到，句子质量和未出现实体两个特征对摘要效果有明显提升，说明这两个特征均有效。同时增加了两个特征后，对整体效果提升更加明显。通过观察可以发现，句子质量对长文本结果的提高更加明显；未出现实体对短文本的提升更有效果。

3.5.2.4 摘要效果举例

摘要例子如表 3-6 所示，其中人工摘要为人工根据所有答案选取的句子，自动摘要为 KSSU 算法抽取出的摘要，没有任何的顺序调整。可以看到自动摘要的效果非常好，涵盖了人工摘要所有的句子，并且有所补充，而且可读性非常强，非常有逻辑性。说明 KSSU 算法效果非常好。

表 3-5 Yahoo answer 不同特征叠加结果

评价指标	KS	KS+SQ	KS+UE	KSSU
ROUGE1_P	48.631	48.566	49.982	**50.286**
ROUGE1_R	56.518	56.752	65.971	**67.148**
ROUGE1_F	50.279	50.393	55.598	**56.244**
ROUGE2_P	32.537	32.824	34.495	**34.855**
ROUGE2_R	39.278	39.604	47.217	**48.264**
ROUGE2_F	34.167	34.505	38.986	**39.614**
ROUGE-SU4_P	33.099	33.295	34.517	**34.800**
ROUGE-SU4_R	39.623	39.915	47.139	**48.069**
ROUGE-SU4_F	34.547	34.820	38.932	**39.468**

3.6 本章小结

本章将答案摘要问题转化成了一个整数规划问题，通过求解整数规划最优解的方式求得摘要。对实体摘要进行两方面的衡量，综合考虑了实体在答案中的重要程度和实体与问题的相关度两方面。对低质量的句子进行过滤，对冗余句子进行句子压缩，从而得出了较为优秀的结果。进一步考虑了 CQA 文本的特殊性，加入了句子质量函数，对高质量的句子有一个加分，从而尽可能选出高质量的句子。考虑到实体扩展可能有一部分实体不能命中，提出了一种针对未命中实体权重衡量的方法，实验结果证明对摘要效果有提高。提出了 KSSU 算

法，比传统基线方法的 ROUGE1 和 ROUGE2 两个指标在 F 值上都高出了近 15 个百分点。通过观察摘要效果实例可以发现，KSSU 算法抽取的摘要可读性非常强，有着很好的效果。

表 3-6 自动摘要举例

问题：How can I get my girlfriend to be more open, kind and outgoing?
最佳答案： I feel you pain brother. The proble could be one of the following: I agree with the previous 2 answers. You can't change anyone. Personality theory states that you are who you are as well as you girlfriend's ability to be intimate or open. Also, I can't help but wounder, how insecure are you in this relationship (do you believe that she is really invested in the relationship as much as you?). I don't mean to sound mean. But when we are insecure, we are hyper-sensitive and needed. Now on the other hand, I've had friends who were girls, who had the dating rule, that you love the guy less, so that you stay in control. Overall, it should like she is not giving you what you want. Life is too short, don't waste your time on her, move on.
人工摘要： If you need to change those qualitys than most likely she isn't the person for you. You cant change her and you shouldnt try either she is the person you fell in love with. she may care for you just not in that empathic kind of way.
自动摘要： I agree with the previous 2 answers. If you need to change those qualitys than most likely she isn't the person for you. You cant change her and you shouldnt try either she is the person you fell in love with. Or perhaps, she just doesn't feel that "close" to you. I mean, she may care for you just not in that empathic kind of way. She isn't seeing things from your point of view YET. are you seeing things from hers?

第 4 章 答案摘要系统设计与实现

4.1 引言

用户在互联网上检索问题一般很难得到答案。CQA 系统如百度知道等为用户提供了一个可以直接搜索想得到问题的社区平台，在社区上用户可以进行提问，由其他用户回答。由于用户水平不一，答案中难免存在有错误的情况，并且由于是开放社区，每个人都可以直接进行回答，因而答案数量可能比较多，用户很难直接从中获取想要的信息。一个比较好的方式是对这些答案进行摘要，得到一个完整、通顺、可读的摘要，补充最佳答案所缺失的信息。本章根据前两章的算法，实现了一个基于社区问答的答案摘要系统。4.2 节介绍了算法的整体流程和具体的实现策略，总结了各个算法在整体流程中的具体作用，具体讨论了如何避免出现多种策略所带来的代码冗余问题；4.3 节介绍了系统整体的设计和架构，将答案摘要系统主要分成爬虫、检索和摘要三个模块，对系统中三个模块都有比较细致的分析，分析了各个模块的具体作用、实现策略和实现方式；4.4 章进行系统的结果展示，通过实际例子证明摘要算法的实际价值；4.5 节总结之前各节的内容。

4.2 答案摘要算法总体流程与设计思想

图 4-1 为 KSSU 算法的具体流程，先读入数据，对句子进行压缩。对清理过的问题和答案进行概念映射。在得到问题和答案的实体后，对问题进行实体扩展，根据问题中的实体得到命中实体，估计实体的权重。然后过滤句子，去掉无关信息的句子。在给句子和实体标号后，建立优化函数，通过工具包 pulp 得到整数规划的最优解，得到摘要句子，然后返回摘要。

摘要算法已放在全世界著名开源网站 github[8]上，采用 MIT License，现已开源。

4.2.1 知识图谱的处理方法

知识图谱选用 ConceptNet5[32]，由于 ConceptNet5 中存在着大量的其他语种的实体，比如日语、汉语、西班牙语等。本课题主要处理英语问题的答案摘要，

[8] https://github.com/lavizhao/insummer

这些实体对于英语的答案摘要是没有任何帮助的，因而需要去掉，经过筛选后的实体关系对共有八百万个，关系有约 30 组。

由于实体扩展算法需要大量的访问知识库，因而对知识库存储上的优化是比较必要的。知识库的存储采用 MongoDB[45]进行存储，并分别对实体，实体实体关系对建立索引，并加以封装。经过优化后，单词查询的时间控制在了 1-10ms 级别。大大的加快了实体扩展的实现速度。选用 MongoDB 而不是比较通用的 MySQL 理由为，MongoDB 的内置缓存是默认开启无穷大的，而 MySQL 在默认访问时只有 1M 大小，因而 MySQL 会比较慢。

在概念映射阶段，需要进行大量的查找知识库中有没有该实体的操作，对 IO 性能要求比较高，因而直接抽取知识库中的词表，将这些词表直接在初始化阶段载入到内存中，加快访问速度。

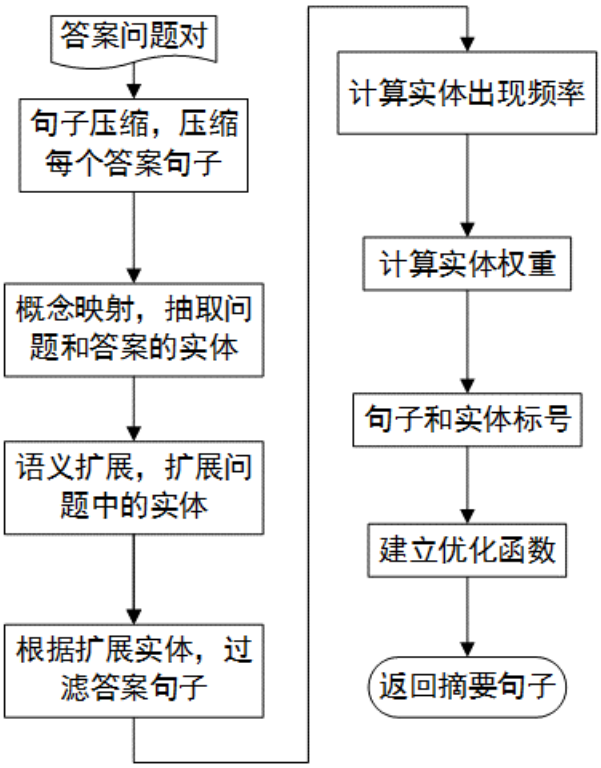

图 4-1 基于知识图谱的答案摘要算法流程

4.2.2 问句实体扩展模块设计思想

实体扩展模块主要处理问题和答案句子的概念映射、句子过滤、实体扩展

以及衡量命中实体。各部分设计思想如下。

在概念映射阶段，首先使用 NLTK[46]工具包对句子进行分句、词性标注以及去停顿词等工作。按照表 2-1 所示的词性转换规则，对所有的词性进行词性转换。然后按照算法 2-1 所示进行概念映射。在句子过滤阶段，按照表 3-1 提供的规则进行过滤。

在实体扩展阶段如果只使用2.3节中介绍的实体扩展方法和2.4节的实体约减方法，实现起来是非常容易的。但是要支持更多如2.5.3节中介绍的约减方法，不使用工厂模式是非常困难的。如图 4-1 所示，图中的类为实体扩展各个策略类及其策略、作用和继承关系。对于实体扩展，实现了一个基类，abstract_entity_expansioner，这个类的功能是对一些通用基础函数的实现，如获取问题的实体、抽取答案句子的所有实体等函数。这个函数有两个抽象方法，expand 函数用来进行实体扩展，run 函数控制整个类的全部流程，其子类必须实现。在子类实现时，只需实现这两个函数的功能即可。

表 4-1 实体扩展各个类作用

类名	策略及作用	父类
abstract_entity_expansioner	抽象类，定义公共函数	无
level_filter_entity_expansioner	层次扩展方法的基类	abstract_entity_expansioner
OnlySynExpansioner	只进行关联层扩展	level_filter_entity_expansioner
SynRelateExpansioner	进行同义层、关联层扩展	level_filter_entity_expansioner
SynPagerankExpansioner	同义、关联层扩展，Pagerank 约减	level_filter_entity_expansioner
SynHitsExpansioner	同义、关联层扩展，hits 约减	level_filter_entity_expansioner
SynCCExpansioner	同义、关联层扩展，连通度约减	level_filter_entity_expansioner
SynKCoreExpansioner	同义、关联层扩展，最小度约减	level_filter_entity_expansioner
RankRelateFilterExpansioner	同义、关联层扩展，Pagerank、启发式约减	level_filter_entity_expansioner

由于这些算法具有共同的特性，基本流程均是先进行同义层扩展，再进行同义层约减(有一些不需要)，然后进行关联层扩展，在关联层扩展的基础上进行约减。因此 level_filter_entity_expansioner 类实现了这些功能，这个类将整个实体扩展的流程拆解成了四个部分：1.同义层扩展；2.同义层约减；3.关联层扩展；关联层约减。syn_expand，syn_filter，relate_expand，relate_filter 四个函数

分别实现了上述功能。

在上述功能均有实现的基础上，扩展出了六个类，分别对应着不同策略，每个策略实现不同的扩展、约减方法，从而得出最后的结果。其中对于 Pagerank、HITS、CC 和 KCore 策略，仍然按照上述思想，实现了一个 ranker 的父类，子类重现这些排序方法，对实体权重进行衡量。Pagerank 和 HITS 算法均使用 networkx[47]工具包，参数选择经典参数。

4.2.3 摘要模块设计思想

摘要模块主要的功能为找出命中实体和未命中实体，对句子进行过滤，衡量命中实体、未命中实体的权重和句子质量，对选中的实体和选中的句子进行标号，做整数规划的预处理，导入到整数规划的工具包内进行优化，得到最优句子集合。摘要算法的整体算法图如图 4-1 所示，实体扩展为其中的一个环节。

首先摘要模块接收实体扩展模块得到的扩展结果，求得命中实体和未命中实体，过滤掉不需要的未命中实体。根据这些实体和句子长度，对无关句子进行过滤。对剩下的实体和句子进行标号，整数规划的输入，导入到整数规划工具包，得到优化结果。

在设计上，也采取了基类定义基础方法，子类实现具体策略的方法。这样做的好处是可以多次重用基础方法，而在具体策略上调整灵活。

4.3 答案摘要系统设计与实现

答案摘要系统实现需要以下几个模块：1. 爬虫模块；2.问题检索模块；3. 摘要模块。系统结构如图 4-2 所示。具体流程为：爬虫模块定期对 Yahoo answer 网站进行爬取，获得的资源经过整理后放到问答资源数据库中，并对其进行索引；在用户搜索问题时，根据倒排索引的内容，进行匹配，使用 BM25 相似度函数，选出最接近的 20 个问题，并加以展示；用户在选中某问题时，系统开始进行答案摘要的抽取工作，在抽取工作完成后，展示问题描述、最佳答案和答案摘要。

4.3.1 爬虫模块的设计与实现

爬虫模块负责定期对 CQA 网站进行爬取，在这里还没有实现定期爬取的功能，主要是通过人工按照规定的时间执行爬虫程序，对 CQA 网站进行爬取，然后哈希去重复。

爬虫模块的整体结构如图 4-3 所示。爬虫一开始选择几个合适的初始种子点，在这里，可以是一个种子点也可以是多个种子点。爬虫的初始点一般选择每个类别的下拉列表，如图 4-4 所示，每个类别下的问题都有很多个，有的是已经解决的有的是没有解决的。对页面中的每个 URL 进行解析，判断是不是一个问题的 URL，问题的 URL 一般都明显带有 URL 字样，因而可以得到该页面的所有 URL 列表。将 URL 列表与已经存在内存中的哈希表进行比对，如果发现已经存在于哈希表总，则说明已经爬取过该问题，则不应该进行爬取，如果没有，则爬取该问题。由于大规模的爬取某个网站是会面临着被封 IP 的问题的。因而在实际应用中，使用浏览器代理，能有效的防止爬取过程中被封 IP 的问题。在爬下来问题后，对 HTML 页面进行解析，抽取到有用的问题，存入到文本文件中即可。

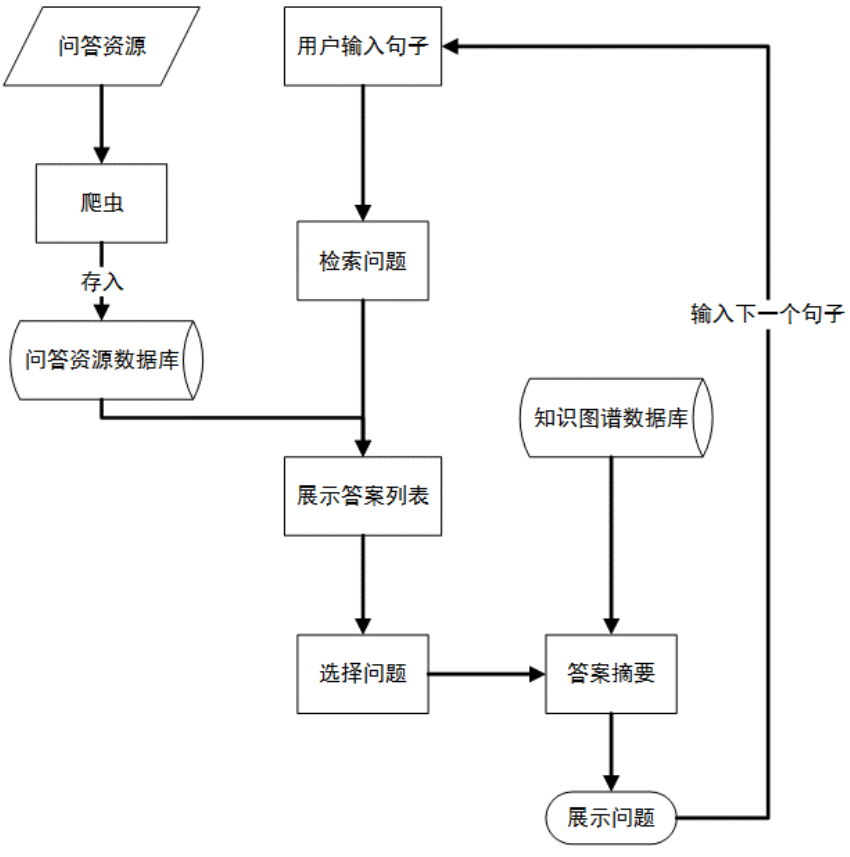

图 4-2 系统结构图

爬虫工具包采用 scrapy[48]工具包，对 URL 进行解析，然后处理 HTML 网页，使用 BeautifulSoup 工具包去除 HTML 页面内的 tag。现阶段已经爬取的网

页内容有 25 万个问题，80 万个答案，数据文件大小 1.5GB。

4.3.2 检索模块的设计与实现

问题检索模块使用 Whoosh[9]工具包建立索引和查询，Whoosh 是一个基于纯 python 的检索工具包，相较于著名开源搜索工具 Lucene[10]，有着使用简单，与 python3 融合好的优点。文档相似度函数采用 BM25[49]，相较于传统的 TF-IDF 方法，BM25 已经被成功应用于检索领域，尤其是对短查询词有着良好的效果。在建立索引后，单次查询时间不超过 0.3s，查询效果优异。

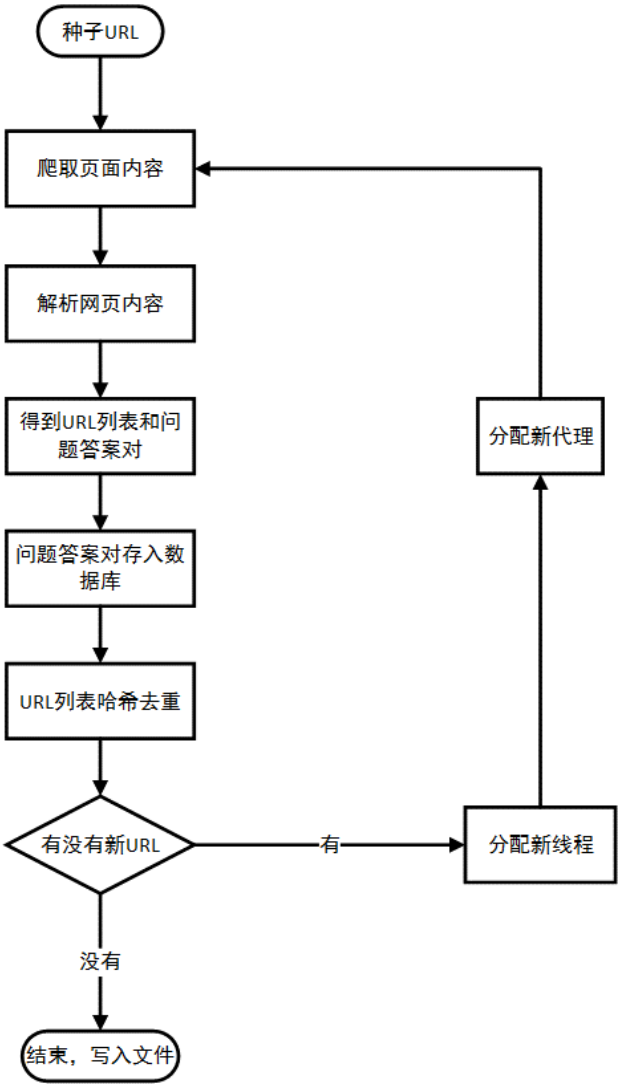

图 4-3 爬虫架构图

[9] https://pypi.python.org/pypi/Whoosh/
[10] https://lucene.apache.org/

4.3.3 摘要模块的设计与实现

摘要方法采用 KSSU 算法，参数选择与 Yahoo answer 数据集上的参数保持一致。对短文本有着良好的筛选效果。在这里为了保证运算时间会比较少，需要采取一些优化措施。比如 MongoDB 的特性是内存缓存数量非常大，近似于等于内存，如果可以提前将 MongoDB 中的部分数据载入到内存中，则会增加程序的执行速度。一个可行的解决办法是拟定一些初始的实体，对在载入系统之前对这些实体进行随机数据库访问，将这些实体的内容直接加在到内存中。另一方面，虽然这些对象存放在内存中，但是 python 通过 MongoDB 的 python 接口每次访问数据时，仍然会新建大量的字符串对象，而由于数据库没有写操作的问题，因而可以在答案摘要内部设置一个哈希表，对查询的 query 进行缓存，可加快程序的实际执行速度。

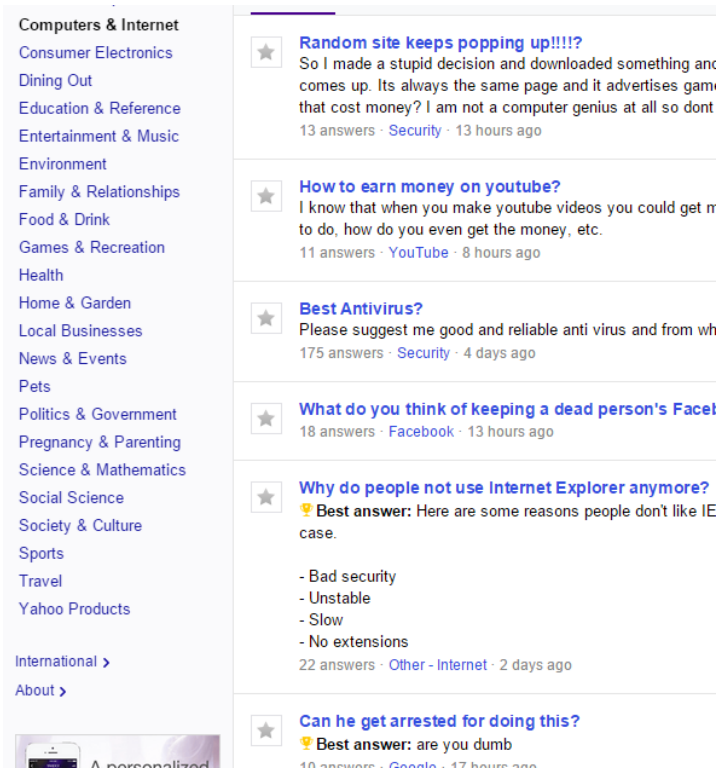

图 4-4 Yahoo answer 每一类的问题列表

4.4 答案摘要系统结果展示

摘要系统主要分为开始界面、问题检索界面和答案展示及摘要界面三部分

组成。系统使用 flask[11]作为 web 框架，flask 相较于 django 和 tornado 有轻便快捷易于使用等特点，尤其是其与 python3 融合比较好。界面由 bootstrap 进行美化。摘要系统开始界面如图 4-5 所示。在开始界面，用户可以输入想要搜索的问题，根据用户搜索的问题，搜索引擎在后台查找出最相似的几个问题，并返回，构成了问题检索界面，如图 4-6 所示。在检索界面，根据用户 query 的相关度进行排序，得到最相关的几个问题。每个链接都可以直接进行访问，在访问的过程中，答案摘要系统开始抽取答案摘要，返回问题的所有答案、问题描述、问题内容和答案摘要结果。

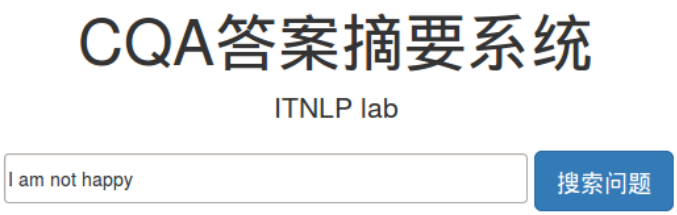

图 4-5 CQA 答案摘要系统开始界面

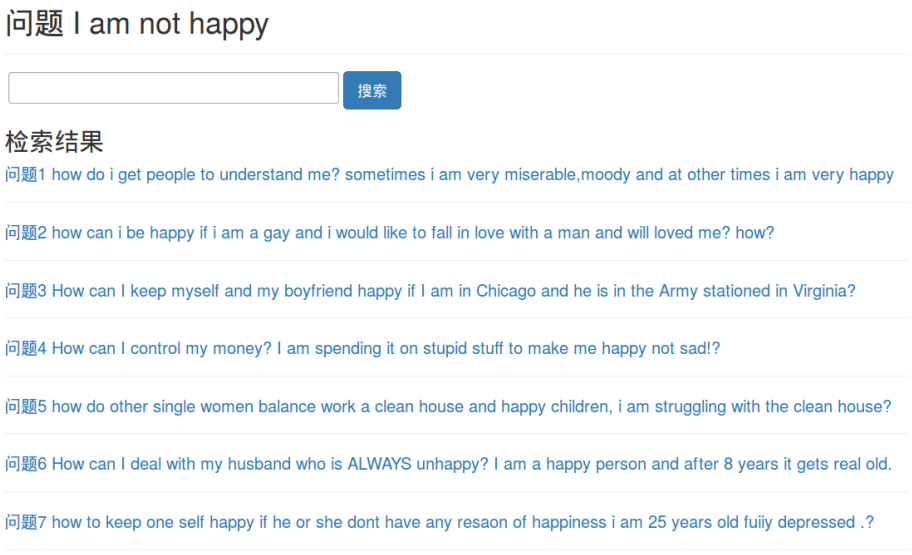

图 4-6 问题检索界面

问题摘要及最佳答案展示如图 4-7 所示。问句主要询问为什么计算机不供

[11] http://flask.pocoo.org/

电的情况下，计算机内的时钟仍然可以工作。这个问题的正确答案是主板(motherboard)内有一块电池(battary)，该电池在不供电的情况下，仍然可以维持RTC 芯片计数。可以看到答案摘要准确的找到了这些实体所在的句子，与问句非常相关，且比最佳答案短，说明摘要的效果是非常有效的。

另一个例子如图 4-8 所示，该问题主要询问怎么将 Itunes 的文件夹从一台电脑移动到另一台电脑上。这个例子的所有答案得出的结果是用个硬盘将数据拷贝过去，而我们看到 4-8 的最佳答案仍然说了不少与问题无关的结果，但是答案摘要则避免了这个问题，说明答案摘要的结果优于最佳答案。

所有答案的检索结果如图 4-9 所示，在检索结果中，可以看到其他用户也提出了一些建议，而最佳答案对于提问者自身的状况也存在着分析不足的情况，答案摘要成功的加入了一些其他用户的意见，补充了最佳答案，因而得到的效果比最佳答案好，且没有出现冗余句子，整体摘要读起来十分顺畅，没有出现传统摘要方法中的句子不连贯的问题。因而可以说，摘要效果是十分理想的。

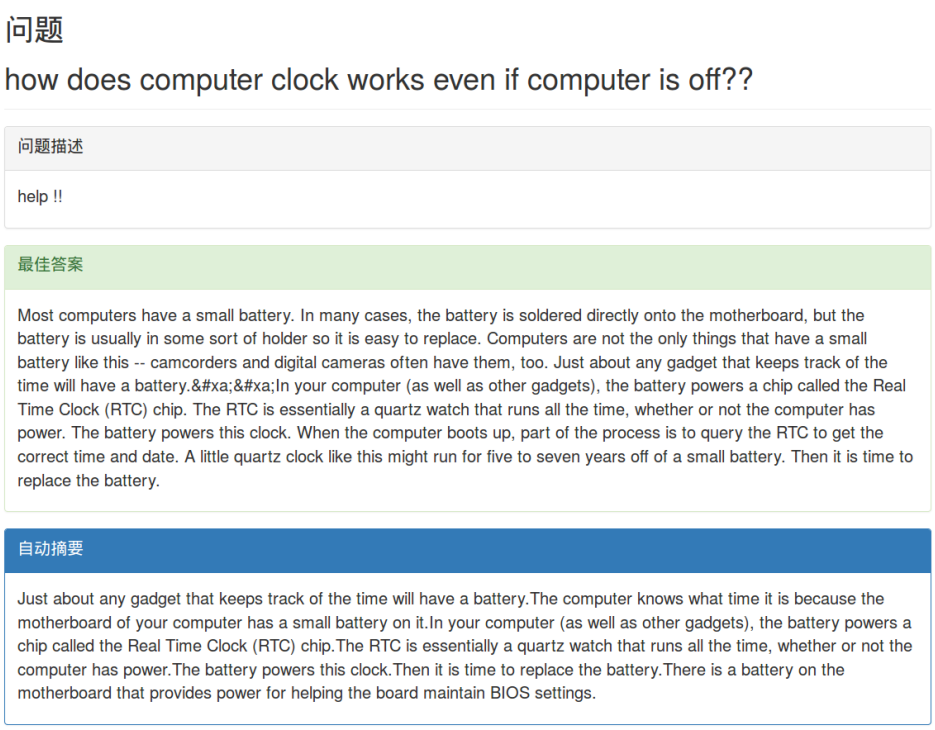

图 4-7 问题最佳答案及答案摘要结果

4.5 本章小结

本章主要介绍了，基于知识图谱的答案摘要系统的实现过程。从摘要算法

的策略设计开始，详细的介绍了答案摘要算法中如何处理多种不同策略的代码冗余问题。然后介绍了答案摘要系统具体的几个模块，分为爬虫、检索和摘要三个大部分组成，介绍了每一个模块具体的工作原理，工作流程。最后引出了答案摘要系统的整体结果，通过举例子的方式，演示了答案摘要系统是如何工作的。通过例子证明了答案摘要方法的效果比最佳答案好，而且没有冗余句子，也没有句子不通顺的情况，因而具有良好的泛化效果。

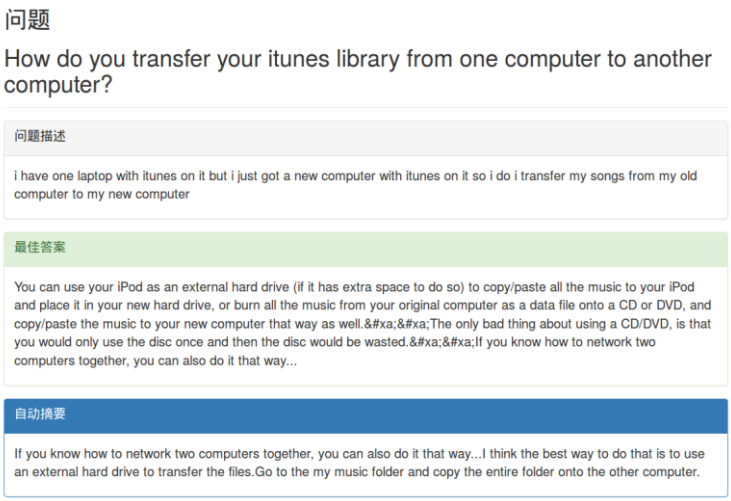

图 4-8 答案摘要例子

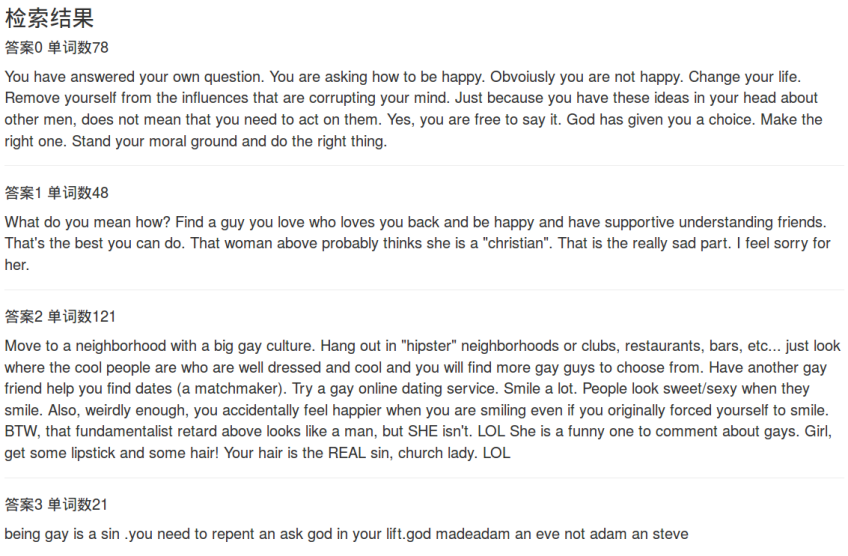

图 4-9 问题所有答案

结 论

答案摘要对提高 CQA 服务有重要作用。答案摘要可以将每个问题下的答案句子加以整合，可以得到了一个完整、流畅、与问题相关的摘要，增强答案的可读性，提高问答对的质量，为 CQA 其他任务提供关键性的帮助。现阶段缺乏一种既可以使摘要和问题相关联，通用性又比较好的方法。本文提出了一种基于知识图谱问句实体扩展的答案摘要方法，从问题出发，对问题做实体扩展，通过扩展的实体对答案句子加以过滤，使用全局算法进行摘要，力图得到一个完备、流畅、与问题相关的答案摘要。本文的主要贡献有：

1. 完成了 CQA 中问句和答案的实体映射方法。CQA 中问句和答案与对话文本和正式文本关注的实体领域不同，其中含有大量的常识类实体，因此使用传统的实体链接方法是不适用的。本文提出了一种基于词性标注的方法，结合常识类知识库 ConceptNet，得到了问句和答案句子的核心组成实体。

2. 提出了 CQA 中问句的实体扩展、约减方法和实体权重计算、评定方法。为了建立问句和答案句子之间的联系，本文借鉴了信息检索领域中的查询扩展方法，提出了基于常识类知识库的实体扩展方法，并与查询扩展方法加以比较。为了增加说服力，本文先后采用了 Pagerank 和一种基于启发式规则的方法，对扩展实体数目进行权重评定和约减。提出了覆盖率和命中实体数目两种指标来衡量扩展实体数目的好坏。通过对扩展算法的评估和下一步摘要算法的结果表明，基于知识图谱的问句实体扩展方法是有效果的。

3. 提出了一种基于问句实体扩展和整数规划的全局摘要抽取算法。本文将实体扩展出来的权重和实体在所有答案中出现的频率相融合，得到了实体的综合权重。利用知识图谱扩展出的实体，设定目标函数，利用整数规划的方法进行抽取式摘要。并分别在两个数据集上进行实验，取得了非常好的效果。更进一步地增加了未命中实体在摘要结果中的影响，提出了一种对未命中实体的权重估算方法，提高了实验结果。

4. 对摘要算法增加了一种对句子质量的估算方法，并与基于知识图谱实体扩展的全局摘要算法想融合，提出了 KSSU 算法。由于高质量的句子含有较多的实体信息，即含有较多的实体，因而本文主要从句子的实体数和句子中所有实体的权重和两方面来考虑答案句子的质量因素，提出了一种对于答案句子质量的衡量函数，并加入目标优化函数，得到了基于句子质量和未命中实体的实体扩展摘要算法，即 KSSU 算法。通过实验证明，KSSU 算法对于短文本和长

文本的效果都高于传统方法。

5. 完成了在一个 CQA 上的答案摘要系统。本文建立了一个基于 CQA 的答案摘要系统，系统主要有两部分组成：答案检索系统和答案摘要系统。通过答案检索系统，用户可以检索到与该用户提出的问题最相近的几个问题，用户可以加以选择；通过摘要系统，用户可以看到问题和问题的描述，以及该问题下的最佳答案、所有答案和系统经过计算得出的答案摘要，可以从多方面让用户更快的获得信息。

本文虽然在答案摘要领域取得了一定的成果，但是由于时间所限，还有一定的提高余地。本文工作的后续可以提高的内容如下：

1. 增加 ConceptNet 知识库的内容，加入 Yago 和 DBpedia 等知识库里面的实体及其之间的关系，并对所有权重进行重新衡量，将全新的知识库用在算法中。

2. 由于现阶段没有成熟的中文知识图谱，本论文没有对中文问题进行答案摘要。在后续研究会探讨中文答案摘要的可能方法。

参考文献

[1] 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.

[2] Berger A, Mittal V O. Query-relevant summarization using FAQs[C]// Proceedings of the 38th Annual Meeting on Association for Computational Linguistics. Association for Computational Linguistics, 2000: 294-301.

[3] Mihalcea R, Tarau P. TextRank: Bringing order into texts[C]//. Association for Computational Linguistics, 2004.

[4] Erkan G, Radev D R. LexRank: graph-based lexical centrality as salience in text summarization[J]. Journal of Artificial Intelligence Research, 2004, 457-479.

[5] Page L, Brin S, Motwani R, et al. The Pagerank citation ranking: Bringing order to the web[J]. 1999.

[6] Carbonell J, Goldstein J. The use of MMR, diversity-based reranking for reordering documents and proDUCing summaries[C]// Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval. ACM, 1998: 335-336.

[7] Wang B, Liu B, Sun C, et al. Adaptive maximum marginal relevance based multi-email summarization [M]. Artificial Intelligence and Computational Intelligence. Springer. 2009: 417-424.

[8] Chali Y, Hasan S A, Joty S R. A SVM-based ensemble approach to multi-document summarization [M]. Advances in Artificial Intelligence. Springer. 2009: 199-202.

[9] Li Y, Li S. Query-focused Multi-Document Summarization: Combining a Topic Model with Graph-based Semi-supervised Learning[J]. 2014.

[10] Ouyang Y, Li W, Li S, et al. Applying regression models to query-focused multi-document summarization[J]. Information Processing & Management, 2011, 47(2): 227-237.

[11] Mcdonald R. A study of global inference algorithms in multi-document summarization[M]. Springer, 2007.

[12] Liu Y, Li S, Cao Y, et al. Understanding and summarizing answers in community-based question answering services[C]// Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1. Association for Computational Linguistics, 2008: 497-504.

[13] He J, Dai D. Summarization of Yes/No Questions Using a Feature Function Model[C]// ACML. 2011: 351-366.

[14] Wang L, Raghavan H, Cardie C, et al. Query-Focused Opinion Summarization for User-Generated Content[C]// Proceedings of COLING. 2014: 1660-1669.

[15] Chan W, Zhou X, Wang W, et al. Community answer summarization for multi-sentence question with group L 1 regularization[C]// Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1. Association for Computational Linguistics, 2012: 582-591.

[16] Pande V, Mukherjee T, Varma V. Summarizing Answers for Community Question Answer Services [M]. Language Processing and Knowledge in the Web. Springer. 2013: 151-161.

[17] Liu X, Li Z, Zhao X, et al. Using concept-level random walk model and global inference algorithm for answer summarization [M]. Information Retrieval Technology. Springer. 2011: 434-445.

[18] Tomasoni M, Huang M. Metadata-aware measures for answer summarization in community question answering[C]// Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, 2010: 760-769.

[19] Nastase V. Topic-driven multi-document summarization with encyclopedic knowledge and spreading activation[C]// Proceedings of the Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2008: 763-772.

[20] Collins A M, Loftus E F. A spreading-activation theory of semantic processing[J]. Psychological review, 1975, 82(6): 407.

[21] Qian X, Liu Y. Fast Joint Compression and Summarization via Graph Cuts[C]// EMNLP. 2013: 1492-1502.

[22] Wang L, Raghavan H, Castelli V, et al. A Sentence Compression Based Framework to Query-Focused Multi-Document Summarization[C]// ACL (1).

2013: 1384-1394.

[23] Lin C-Y. Rouge: A package for automatic evaluation of summaries[C]// Text Summarization Branches Out: Proceedings of the ACL-04 Workshop. 2004: 74-81.

[24] Miller G A. WordNet: a lexical database for English[J]. Communications of the ACM, 1995, 38(11): 39-41.

[25] Resnik P. Using information content to evaluate semantic similarity in a taxonomy[J]. arXiv preprint cmp-lg/9511007, 1995.

[26] Ponzetto S P, Strube M. Exploiting semantic role labeling, WordNet and Wikipedia for coreference resolution[C]// Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics. Association for Computational Linguistics, 2006: 192-199.

[27] Kasneci G, Ramanath M, Suchanek F, et al. The YAGO-NAGA approach to knowledge discovery[J]. ACM SIGMOD Record, 2009, 37(4): 41-47.

[28] Bizer C, Lehmann J, Kobilarov G, et al. DBpedia-A crystallization point for the Web of Data[J]. Web Semantics: science, services and agents on the world wide web, 2009, 7(3): 154-165.

[29] Bollacker K, Evans C, Paritosh P, et al. Freebase: a collaboratively created graph database for structuring human knowledge[C]// Proceedings of the 2008 ACM SIGMOD international conference on Management of data. ACM, 2008: 1247-1250.

[30] Havasi C, Speer R, Alonso J. ConceptNet 3: a flexible, multilingual semantic network for common sense knowledge[C]// Recent advances in natural language processing. 2007: 27-29.

[31] Liu H, Singh P. ConceptNet—a practical commonsense reasoning tool-kit[J]. BT technology journal, 2004, 22(4): 211-226.

[32] Speer R, Havasi C. Representing General Relational Knowledge in ConceptNet 5[C]// LREC. 2012: 3679-3686.

[33] Hsu M-H, Tsai M-F, Chen H-H. Query expansion with conceptnet and wordnet: An intrinsic comparison [M]. Information Retrieval Technology. Springer. 2006: 1-13.

[34] Hsu M-H, Tsai M-F, Chen H-H. Combining WordNet and ConceptNet for automatic query expansion: a learning approach [M]. Information Retrieval Technology. Springer. 2008: 213-224.

[35] Kotov A, Zhai C. Tapping into knowledge base for concept feedback: leveraging conceptnet to improve search results for difficult queries[C]// Proceedings of the fifth ACM international conference on Web search and data mining. ACM, 2012: 403-412.

[36] Hoffart J, Yosef M A, Bordino I, et al. Robust disambiguation of named entities in text[C]// Proceedings of the Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2011: 782-792.

[37] Han X, Sun L, Zhao J. Collective entity linking in web text: a graph-based method[C]// Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval. ACM, 2011: 765-774.

[38] Bunescu R C, Pasca M. Using Encyclopedic Knowledge for Named entity Disambiguation[C]// EACL. 2006,6: 9-16.

[39] Schnabel T, Schütze H. Flors: Fast and simple domain adaptation for part-of-speech tagging[J]. Transactions of the Association for Computational Linguistics, 2014, 215-26.

[40] Hu M, Liu B. Mining and summarizing customer reviews[C]// Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2004: 168-177.

[41] Cao X, Cong G, Cui B, et al. A generalized framework of exploring category information for question retrieval in community question answer archives[C]// Proceedings of the 19th international conference on World wide web. ACM, 2010: 201-210.

[42] Takamura H, Okumura M. Text summarization model based on maximum coverage problem and its variant[C]// Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics. Association for Computational Linguistics, 2009: 781-789.

[43] Li C, Liu F, Weng F, et al. Document Summarization via Guided Sentence Compression[C]// EMNLP. 2013: 490-500.

[44] Galley M, Mckeown K. Lexicalized Markov Grammars for Sentence Compression[C]// HLT-NAACL. 2007: 180-187.

[45] Wei-Ping Z, Ming-Xin L, Huan C. Using MongoDB to implement textbook management system instead of MySQL[C]// Communication Software and Networks (ICCSN), 2011 IEEE 3rd International Conference on. IEEE, 2011: 303-305.

[46] Bird S. NLTK: the natural language toolkit[C]// Proceedings of the COLING/ACL on Interactive presentation sessions. Association for Computational Linguistics, 2006: 69-72.

[47] Schult D A, Swart P. Exploring network structure, dynamics, and function using NetworkX[C]// Proceedings of the 7th Python in Science Conferences (SciPy 2008). 2008,2008: 11-16.

[48] Wang J, Guo Y. Scrapy-Based Crawling and User-Behavior Characteristics Analysis on Taobao[C]// Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), 2012 International Conference on. IEEE, 2012: 44-52.

[49] Robertson S, Zaragoza H, Taylor M. Simple BM25 extension to multiple weighted fields[C]// Proceedings of the thirteenth ACM international conference on Information and knowledge management. ACM, 2004: 42-49.

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《基于问句实体扩展和全局规划的答案摘要方法研究》，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：　　日期：　　年　　月　　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：　　日期：　　年　　月　　日

导师签名：　　日期：　　年　　月　　日

致　谢

首先我想对所有帮助过我的老师、同学表示最诚挚的谢意。

我要感谢我的导师王晓龙教授，王老师渊博的学识感染着我，王老师用他的学术热情感染者周围的每一个人。王老师用他诚挚的学术热情、严谨的学风教导着我实验室两年的成长。

我要感谢我的责任导师刘秉权副教授，刘老师耐心的教导、敏锐的学术思维指导着我研究生两年、大四一年的学术生活。在刘老师的手下我得到锻炼，在刘老师的教导下我得到成长，我要衷心的感谢刘老师的深深教诲。

我要感谢实验室的其他老师，如刘远超老师、林磊老师、刘铭老师等，是这些老师在日常的学习和生活中帮助我，鼓励我成长，谢谢你们！

我要感谢实验室的刘峰师兄，徐振师兄等，是你们在每周的学术会议上给我提供了宝贵的指导意见，给我在论文题目、内容上的确定提供了宝贵的意见，还不辞辛劳的帮我修改论文，衷心的感谢你们！

我要感谢寝室的所有小伙伴，如闫铭、姚明、刘金宝同学，虽然在学习上不如我，但是在生活上关心我，为我解决了很多生活上的难题。在日常的课余时间，也同我一同玩耍。

我要感谢实验室的同学，如陈俊文、靳小强、孔行、李思琴和罗菲等，是你们在我开题、中期期间给予了我帮助，在日常的学习中不断给我提供新的方向，谢谢你们。感谢成昊师弟，帮助我完成了一些基础工作。

我要感谢我的女友沈奕聪，是你在我本科和研究生阶段陪伴着我，没有你的压力我不可能成长的这么快，你的幽默风趣是我成长的动力。

感谢我的父母在我二十五年的人生中始终对我加以培养，加以鼓励。

最后，衷心祝福每一个帮助过我的人，愿你们越来越好！