博士学位研究生学科综合考试报告

|  |  |
| --- | --- |
| 院 系： | 信息科学技术学院 |
| 姓 名： | 林 泽 琦 |
| 学 号： | 1401111333 |
| 专 业： | 计算机软件与理论 |
| 研究方向： | 软件工程与软件工程环境 |
| 导 师： | 张 路 教授 |

2013年 3月

# 摘 要

//TODO

# 目录

[摘 要 II](#_Toc477121403)

[目录 IV](#_Toc477121404)

[图表目录 V](#_Toc477121405)

[1. 特征定位技术 1](#_Toc477121406)

[2. 知识表示学习技术 2](#_Toc477121407)

[3. 问答系统 3](#_Toc477121408)

[3.1. 背景 3](#_Toc477121409)

[3.2. 基本概念 4](#_Toc477121410)

[3.3. 问答系统的基本结构 6](#_Toc477121411)

[3.3.1. 问题处理模块 7](#_Toc477121412)

[3.3.2. 段落检索/索引模块 12](#_Toc477121413)

[3.3.3. 答案处理模块 16](#_Toc477121414)

[3.4. IBM Watson 18](#_Toc477121415)

[3.4.1. 信息来源 19](#_Toc477121416)

[3.4.2. 问题分析 21](#_Toc477121417)

[3.4.3. 假设生成 22](#_Toc477121418)

[3.4.4. 假设与证据评分 23](#_Toc477121419)

[3.4.5. 最终答案生成 23](#_Toc477121420)

[3.5. 参考文献 23](#_Toc477121421)

# 图表目录

[图 3‑1 问答系统结构图 7](#_Toc477121719)

[图 3‑2 问题分类及其在TREC数据集上的分布 8](#_Toc477121720)

[图 3‑3基于层次结构分类器的问题分类方法 9](#_Toc477121721)

[图 3‑4 问题模版信息抽取示例 10](#_Toc477121722)

[图 3‑5 语义模版的形式化定义 11](#_Toc477121723)

[图 3‑6 语义模版的使用示例 12](#_Toc477121724)

[图 3‑7 IBM Watson的整体架构 19](#_Toc477121725)

[图 3‑8 定义类型的语料的一个示例 20](#_Toc477121726)

[图 3‑9 对定义类型的语料的重构示例 20](#_Toc477121727)

[图 3‑10 信息来源扩展的流程示例 21](#_Toc477121728)

第一章 特征定位技术

# 特征定位技术

第二章 知识表示学习技术

# 知识表示学习技术

第三章 问答系统

# 问答系统

## 背景

互联网已经成为网络用户们保存数据、交流信息以及共享知识的大平台。如今，大量数字产品网站被创建以满足用户日常生活中的各种需求，网页的数量仍然保持高速无限制的增长。Brin曾如此定义万维网：“万维网是一个大量的完全无法控制的多种多样的文档的聚集地“[Brin 1998]。但是，在如此浩瀚的互联网中，用户如何得知网站的信息并访问这些网站，如何从相关信息服务中提取自己所需要的信息则成为了一大挑战。管理、检索、过滤海量信息的技术，成了一件亟需解决却又非常困难的事情。针对这种情况，广大学者们开始研究为人们在互联网资源中导航并快速访问所需的相关网页的技术。自此，处理海量数据的Web信息检索技术被提出并逐渐被计算机界所重视[Salton 1986]。信息检索，是指将信息按一定的方式组织和存储起来，并根据用户的需要找出有关的信息的过程。

搜索引擎是一种最常见的互联网信息检索技术，其基本思想是：使用自动搜索爬虫来遍历互联网，将互联网上分布的信息下载到本地文档库；然后对文档内容进行自动分析并建立索引；对于用户提出的检索请求，搜索引擎通过检查索引找出匹配的文档并返回给用户。当前，搜索引擎是用户在互联网上寻找所需信息的最主要方式。然而，从与用户交互的方面来看，用户在使用搜索引擎的时候，经常遇到如下的问题：

1. 并不清楚到底怎么样的词语序列才可以返回精确的结果；
2. 可能接收到无关的返回信息；
3. 对搜索引擎通用的布尔逻辑不太理解，更不用说使用高级检索了；
4. 新用户还需要花时间来熟悉搜索引擎的使用方法；
5. 85%的用户只看返回的第一页面，而一般的搜索引擎返回的相关性信息太多，这对排序方法提出了很高的要求。

同时，随着技术发展，用户的需求也不断提高。而关键词的逻辑组合来表达检索需求，难以处理用户复杂的检索要求。在很多情况下，用户所需的信息并不是直接显示表达在互联网文档中的。而传统以关键词为基础的索引、匹配算法停留在语言表层，没有触及语义。那么，如何满足用户进一步的需求，提高检索的准确率以及检索结果的质量，成为现今信息检索领域研究的新课题。要让机器不仅“认识“自然语言文本，还要”理解“自然语言文本；不仅从大规模文档中找出相关文档，还要从相关文档中找出相关的部分。

问答系统是人们探索机器智能的一个应用平台，期望机器可以在一定程度上理解并运用自然语言。面对一个以自然语言表述的问题，一个问答系统的任务就是在给定的文档中找出正确的答案，并以自然语言的表达方式返回答案。从技术的角度看，它是信息检索的一种特殊方式，但这种系统与原先的信息检索系统并不相同。当前的信息检索系统可以做到返回可能含有相关信息的整篇文档，让用户自己从中去寻找答案。从功能上看，信息检索系统旨在在海量的文档中为用户大幅度地降低信息定位的范围；而问答系统则旨在进一步地从中精准地获取问题的答案。从处理技术上看，信息检索系统由于需要处理海量的文档，无法对文本进行深入分析处理，通常只能利用统计及关键词匹配等技术；而问答系统则往往需要借助自然语言处理技术，对文本中的句子或篇章进行深入的语义分析，从而从信息检索系统所返回的粗粒度结果中准确的答案。信息检索系统与问答系统之间的差异，也恰恰体现出了二者之间的互补性。在问答系统的技术发展过程中，信息检索技术发挥了重要的作用。本章将从信息检索系统的视角出发，对基于自然语言的自动问答系统进行综述。

## 基本概念

**问答系统** 关于问答系统的内涵和外延，很多的研究者都给出了各自的定义。例如，[Mollá 2007]将问答系统定义为一个能回答任意自然语言形式问题的自动机。虽然定义很多，并且各种定义之间略有不同，但是一般都认为问答系统是一种智能化的搜索引擎，其输入应该是自然语言形式的问题，输出应该是一个精确、简洁的答案或者可能答案的列表，而不是一堆相关的文档。

**问题** 在问答系统中，一个问题指的是一个清楚地描述了用户的信息需求的自然语言句子。问题一般是一个以疑问词开头的疑问句，有的时候也可能是一个以动词开头的祈使句。相比于信息检索系统以搜索关键词组成的简单列表作为输入，自然语言句子形式的问题中还蕴含有关键词之间的句法结构信息，并能据此体现出其间的语义关联。对于用户提出的自然语言问题，可以将它们大致分为八大种类：事实类问题、列表类问题、定义类问题、假设类问题、原因类问题、关系类问题、过程类问题、确认类问题[Kolomiyets 2011]。以下对这些种类的问题做出简单解释：

* 事实类问题：一般指要求回答实体、时间、地点、数量等事实性内容的问题。例如：“奥巴马的夫人是谁？“；”2014年世界杯的举办地是哪里？“；“水的沸点是多少度？“；等等。
* 列表类问题：与事实类问题的目的相似，但要求的不是一个答案，而是一组答案。例如：“中国有哪些邻国？”；“列出三种原产于澳大利亚的动物”；等等。
* 定义类问题：要求对某个实体进行解释。更具体地，可以分为描述类问题和观点类问题。描述类问题指的是要求给出实体的具体定义；观点类问题则指的是要求给出对实体的看法或评价。
* 假设类问题：指的是问题中陈述了一些假设作为前提约束条件，例如：“如果核战争爆发了，我们应该怎么办？”。
* 原因类问题：要求解释事件原因的问题，一般指以“为什么”开头的问题；
* 关系类问题：要求给出两个实体之间的关系，例如：“以色列和巴勒斯坦是什么关系？”；
* 过程类问题：问题给出了一个任务，要求答案给出解决这一任务所需要执行的具体步骤的列表，例如：“怎么做西红柿炒鸡蛋？”；
* 确认类问题：可以用“是”或则“否”回答的问题，即特殊疑问句。

**信息来源**问答系统的信息来源指的是用于抽取答案的信息对象的集合。

从信息的表示形态方面，问答系统的信息来源可以大致分为**非结构化数据**和**结构化数据**两种[Moens 2006]。非结构化数据指的是诸如自然语言文本、图像、音频、视频等机器难以理解与解释其语义的数据。在问答系统中，非结构化数据在大多数情况下指的是自然语言文本形式的文档集合。结构化数据则指的是数据中的实体及实体间的关联关系得到明确描述，并可以被机器识别与理解的信息来源。典型的结构化数据包括：关系型数据库、专家系统中的三元组知识库、本体网、等等，一般通过专家编辑或是从非结构化数据中进行信息抽取而形成。近年来，互联网上也逐渐出现了一些基于众包编辑的大规模的结构化知识库，如：谷歌知识图谱、DBpedia、Freebase等。相比于非结构数据，结构化数据所表示的语义信息更加明确，更有利于机器处理，但在信息的数量与丰富度上却远远低于非结构数据。目前，从信息来源的维度上看，问答系统的相关研究工作可以划分为以非结构化数据为信息来源的问答系统和以结构化数据为信息来源的问答系统。前者立足于信息检索，主要思想是在海量的自然语言文本文档中搜索出小范围的与问题相关的文档，再从文档中抽取出问题的答案；后者立足于知识表示与推理，主要思想是对非结构化数据中的信息进行结构化表示，并将问题转换为结构化数据上的逻辑语句，并通过逻辑推理求解出问题的答案。

从信息来源的范围方面，问答系统可以分为**限定领域问答系统**与**开放领域问答系统**。限定领域指的是系统能接受的问题只能是关于某个特定的主题的，开放领域指的是系统能接受的问题可以是任意主题的问题，没有任何限制。限定领域的问答系统与基于非结构化数据的问答系统具有很高的重合性：早期的问答系统多为限定领域的问答系统，其信息来源多是由专家编辑构造的结构化数据库或知识库。最著名的限定领域问答系统是BASEBALL[Green 1961]和LUNAR[Woods 1977]。BASEBALL系统能回答关于美国棒球联赛一个赛季的相关问题；LUNAR系统能回答关于阿波罗月球探测任务取回的岩石样本分析结果的相关问题。目前，限定领域的问答系统仍然被大量使用，典型的应用场景是生物医药领域，例如：[Zweigenbaum 2003], [Sang 2005], [Mollá 2007]，等等。此类问答系统又被称为AI系统、专家系统，或是数据库的自然语言接口系统，其相关研究的活跃时期为20世纪60到80年代。近年来，此类技术没有大的突破，因此本文不对其进行详细介绍。进入20世纪90年代，由于互联网的飞速发展，产生了海量领域开放、以无结构自然语言文本为主要形式的文档数据，这为问答系统进入开放领域提供了客观条件。信息检索评测组织TREC自1999年开始每年都设立开放领域问答的评测任务[Dang 2007]，同时其它评测组织如NTCIR和CLEF也设置有开放领域问答系统的评测任务，极大地推动了开放领域问答系统的研究进展。本文将以开放领域问答系统为主线进行阐述。

## 问答系统的基本结构

Moldvan等人在LASSO问答系统中提出，问答系统有三个最基本的组成成分：问题处理模块、段落抽取/索引模块和答案处理模块 （[Moldovan 1999], [Moldovan 2000]）。现有的问答系统大多是以这样的基本模式为基础进行拓展的。图 3‑1展示了问答系统的基本结构图。每一个问题，依次经过三个模块，得到答案返回。问题处理模块通过文本处理技术分析用户新提出的问题，以抽取用户的需求并利用某种表达方式将这种需求信息表达为查询，作为段落抽取/索引模块的输入。索引模块负责从数据库中搜集并管理所有可利用的文档，这些文档可以从互联网中获得，也可以从本地的数据源中获得。段落抽取模块则将索引得到的文档进行分段处理；然后依据这一模块的输入，从中抽取出所有与查询相关的段落；最后，对所有得到的段落进行评估，并将最相关的若干段落作为答案处理模块的输入。答案处理模块则是通过信息抽取技术从这些相关段落中抽取出若干候选答案，通过评估排序获得针对该问题较为可信的答案。

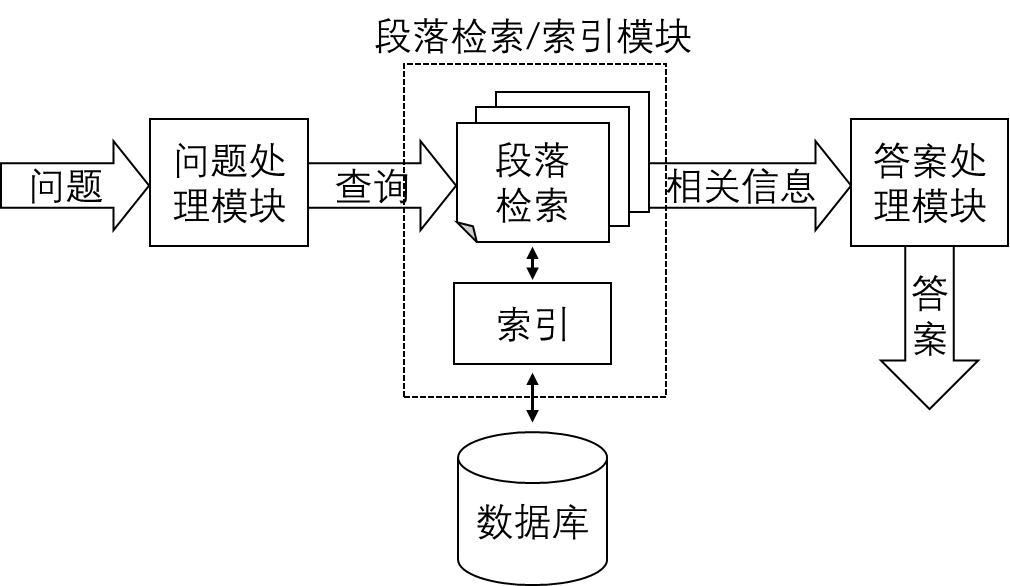


图 3‑1 问答系统结构图

在接下来的三个小节中，我们分别从这三个基本模块出发，分别介绍自动问答系统中各个模块的任务、核心研究点与相关工作。

### 问题处理模块

问题处理模块的主要目标，是找出问题句中对所需答案的约束信息，以帮助后续模块中的信息检索和信息抽取任务的开展。很多工作致力于根据问题句找出期望的答案中可能包含的关键词、答案可能的表达形式等等。例如，给定一个自然语言表达的问题作为输入，在LASSO中，问题处理模块需要完成以下几部分的工作：

1. 确定问题的类型；
2. 确定期望得到的答案的类型；
3. 确定问题的焦点；
4. 得到问题的关键字的集合，作为段落索引模块的查询输入。

#### 问题分类

**问题分类**指根据问题的答案类型对问题进行分类。开放领域的问题可以千变万化，但是同种类型的问题不管是从形式上看还是从内容上看，都是一个相对较小的集合。在较小且有共性的集合中，文本处理的方法就有许多共性。因此，问题分类的制定和问题类型的识别就成为了问题处理模块中最重要的功能之一。目前，大多数问答系统都利用答案类型来指导后续的步骤，尤其是答案抽取策略。例如，对于问人物的问题，答案抽取会利用人物的各种特征来提取答案候选集合。

一般问题分类可以通过疑问词直接决定问题的类型，例如，对于含有“谁”的问题，可以认为需要的答案类型是人物。但这种方法的粒度过于粗糙，难以满足问答系统的实际需求。例如，对于“什么”、“怎么”这样的疑问词，可以对应非常多的答案类型；同时，还有一些问题不包含疑问词。针对这些问题，[Li 2002]提出了更加详细的分类。该问题分类体系中有6个大类（略缩语、描述、实体、人物、地点、数量），在6个打雷下面又分了50个小类。例如，在数量类中，有距离、钱等小类。图 3‑1展示了该问题分类体系，并在TREC数据集中的1000个问题中统计了各个问题类型的分布情况。在问题类别的制定中，某些问题类型的边界一直存在争论，使得一些问题的分类模棱两可。允许同一个问题有多个问题类型标记，可以部分地消除这样的模棱两可[Lampert 2004]。

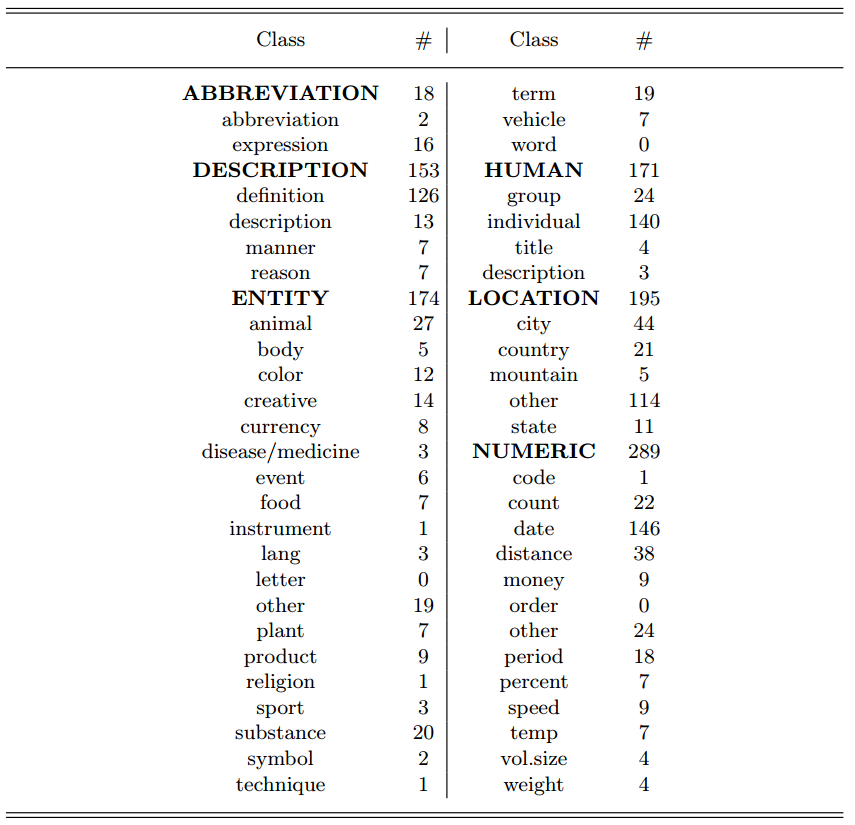


图 3‑2 问题分类及其在TREC数据集上的分布

问题分类的任务就是把一个问题自动划分到已有的分类结构中的一个或几个类。问题分类的方法主要包括模式匹配方法和机器学习方法两类。模式匹配方法为每一种问题类型建立一个模式集合，对于一个问句，只要与某种文体类型对应的模式相匹配，就被认为是这种类型的问题。机器学习方法首先定义一个问题的特征集合，然后在训练数据上得到一个分类器，就可以对新的问题进行分类了。[Zhang 2003]使用的是问题中的表层n-gram特征，并调研了K-近邻算法、决策树、朴素贝叶斯等多种分类方法。在这些分类方法中，支持向量机在问题分类任务上的表现最好。[Li 2002]使用的特征更为深此次，包括语法（词性、词组）和语义（解释、近义词）信息。首先，顶层分类器把问题归类在一个大类别中；然后，根据该大类内的分类器，再将问题具体分到小类别中。这一方法的示意图如图 3‑3所示，在问题分类任务上取得了很好的分类效果。

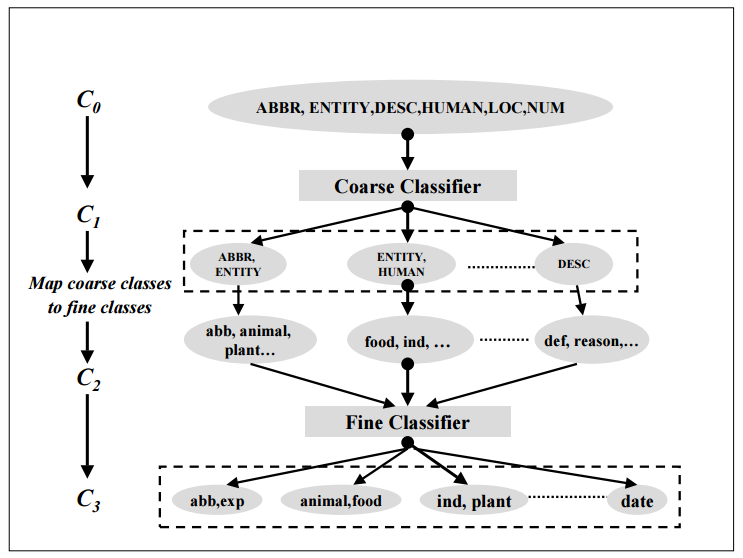


图 3‑3基于层次结构分类器的问题分类方法

#### 问题模版

一般来说，只知道问题的类型不足以找出正确的答案。为了从问题中抽取相关信息以表征用户的需求，在问题处理这一模块中还需要提取**问题模版**。问题模版是一种抽象化的归纳表现形式。在问题分类将具有相似语法结构和相同语义信息的问题归为一类之后，每一类的问题将被映射到对应的模版上，从而可以通过统一的策略进行处理。因此，一个问题模版特征实际上是一种由语法语义信息到问题处理方案的映射信息，任何匹配上该模版的问题都可以依照该方案进行处理。现有系统大多通过频繁结构抽取语法及语法约束，为每一类问题构建一个问题模版，然后为该模版制定若干答案抽取模式。当系统接收到一个新问题的时候，首先会通过模版匹配搜寻到该问题的问题模版，而后得到该问题所对应的答案模版，最后利用答案模版从文档集中抽取出该问题的答案。例如，对于问题：“谁是第一个实现太空行走的俄罗斯宇航员？”（“Who was the first Russian astronaut to walk in space?”），[Harabagiu 2000]首先根据疑问词Who识别出该问题是一个PERSON类的问题，随后从该问题的句法树中抽取信息，以映射到问题模版上。图 3‑4对该问题的句法树以及最终映射到的问题模版的信息进行了展示。

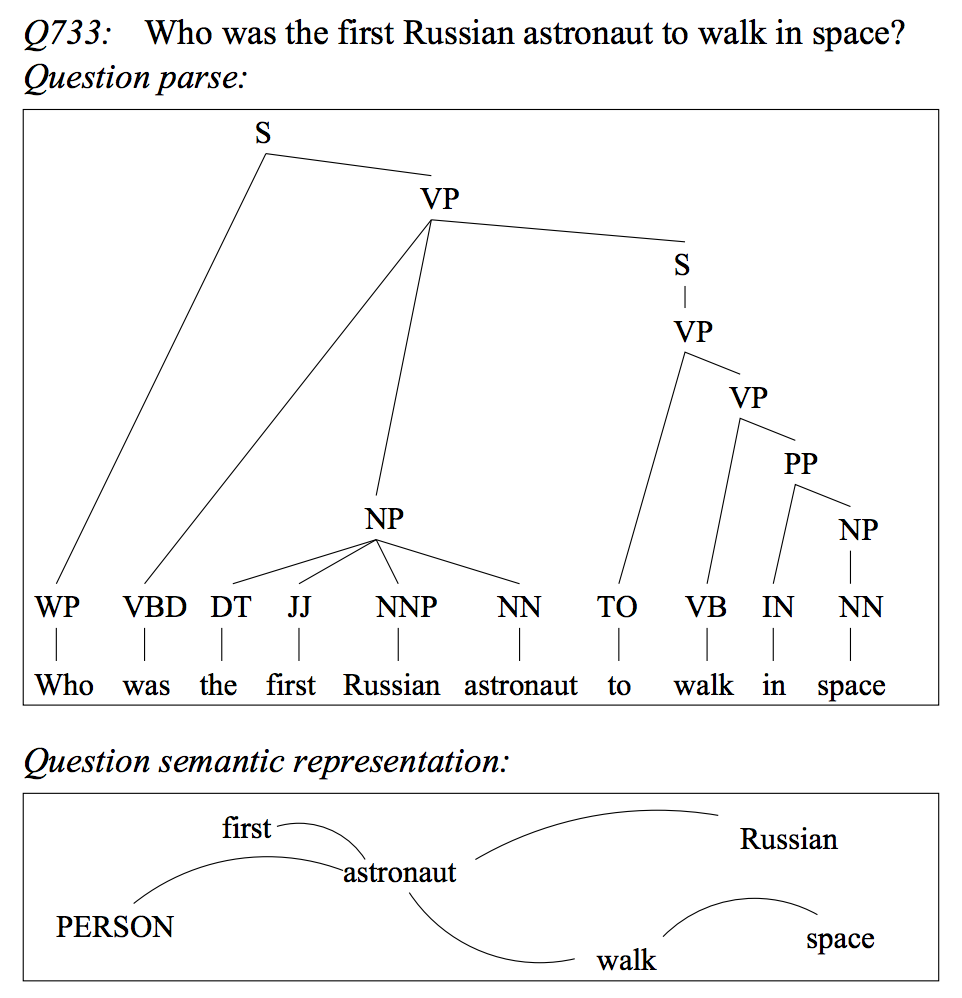


图 3‑4 问题模版信息抽取示例

早期的问答系统中，问题模版多为浅层模版。浅层模版只是对语法结构作出限制，但是不具有语义约束信息，不能够区分结构相同但语义不同的问题。然而，很多情况下具有相同结构的问题却有不同的语义，而且抽取答案的策略也截然不同。因此，随着技术的发展，问答系统逐渐使用语义模版来取代浅层模版。语义模版是在浅层模版的基础上添加语义约束信息，将浅层模版进一步细化。语义模版用一些语义标签将覆盖的问题集进一步分割成不同的子集；而后针对每一个子集，再分别制定答案模版，这样痛过分情况处理的方法保证了答案的精确性。[Hao 2008] 给出了语义模版的形式化定义，如图 3‑5所示。模版中的核心元素包括：问题目标、问题类型、概念、事件、约束。图 3‑6给出了这一形式化定义下的语义模版的一个使用示例。

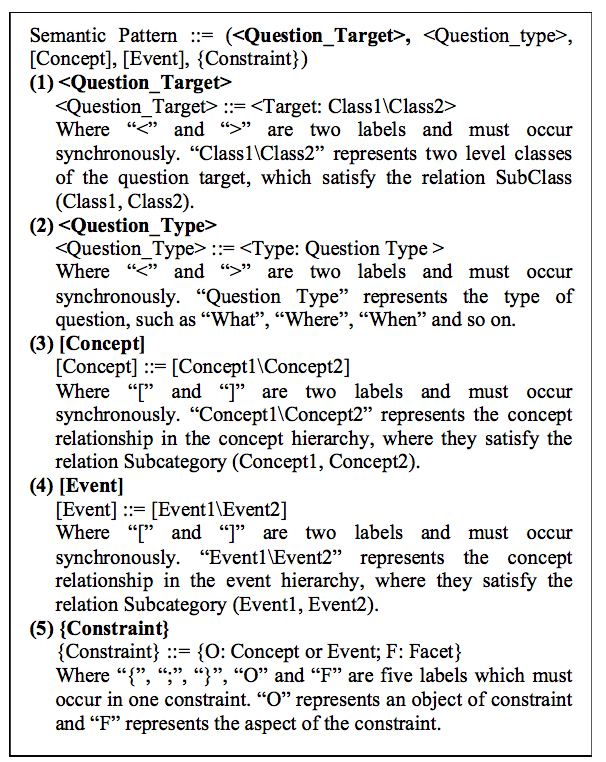


图 3‑5 语义模版的形式化定义

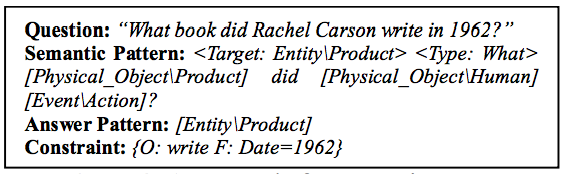


图 3‑6 语义模版的使用示例

有了语义模版，就需要通过**语义关系抽取**，自动地把每个问题中所蕴含的用户需求通过若干有语义信息标注的词条表达出来。经过语义信息抽取过程，每个问题从初识的语法结构约束中脱离出来，在统一的语义空间中重新定位。现有的语义关系抽取技术大都是以自然语言处理结果为基础的，例如：基于依存树分析的方法（[Liu 2007], [Bouma 2002]）；基于语义角色标注分析的方法（[Carreras 2005], [Hacioglu 2004]）；基于规则分析的方法（[Hu 2007]）；等等。

### 段落检索/索引模块

段落检索/索引模块的主要目的是缩小答案的范围，提高下一步答案抽取的效率和精度。

缩小答案范围的最简单方法是去掉问题中的停用词和问句相关的词（如疑问词），生成查询，然后利用已有的信息检索模型进行检索，把返回的结果作为答案提取部分的输入。这种方法很难获得较好的效果，而文档检索的效果会直接影响到问答系统的整体性能[Collins 2004]。如果检索系统的相关性精度较差，那么会有大量无关文档需要后续处理，而答案提取通常采用复杂的自然语言处理技术，这必然导致系统整体效率低下。如果检索系统的召回率较低，那么很多包含答案的文档或者段落没有被返回，显然含有正确答案的文档或者段落越少，提取出正确答案的可能性也越小，这会导致系统整体性能较差。

#### 关键词提取

为了找到与问题相关的文本文档集合，首先需要对问题进行**关键词提取**，以作为信息检索的输入。最简单的方法是删除问题中的停用词，将其余的词作为关键词，并使用TF-IDF等统计学方法来度量每个关键词的重要程度。在此基础上，有一些研究工作旨在通过提高关键词提取的质量，从而提升问答系统的精度。一方面，随着WordNet和HowNet等同义词典的应用，可以从语义层面对关键词集合加以拓展。[Dave 2003]利用WordNet等语义词典获取与关键词集合中的词条的语义相近的其它词条，再将这些新得到的词条加入到关键词集合中，这样可以解决同义异构的问题，有效地挖掘出潜在的答案，提高了答案抽取的召回率；另一方面，考虑关键词数量对信息检索系统的影响：当关键词数量很多时，检索返回的文档较少，召回率就会偏低；当关键词较少时，检索返回的文档偏多，相关性文档的精度就会很低。因此，针对这种关键词数量选择上的不确定性，可以动态地对关键词进行调整。[Moldovan 2000]采用这种迭代式调整技术，多次检索，根据返回文档的多少，调整关键词集合，决定是否增删关键词以及是否采用词形、句法或者语义级别的扩展形式。此外，从答案的角度看：问答系统的目的是找出一个问题的答案，而上述方法是找一个和问题相关的文档。因此，如果从一个问句推测它的答案中可能包含的关键词，用这些关键词来进行查询，会得到更好的效果。对于特定类别的问题，可以从训练数据中学习得到这类问题的回答模式，根据得到的模式从问题中生成包含答案关键词的查询[Agichtein 2004]。在从问题中抽取得合适的关键词集合之后，即可使用信息检索技术获取与问题相关的文档。信息检索领域常用的模型包括布尔模型、向量空间模型、语言模型、概率模型等。实验发现，在问答系统的文档检索中，简单的布尔模型、概率模型与改进的向量空间模型的效果相当（[Moldovan 2003], [Tellex 2003]）。

#### 段落检索

信息检索系统的检索粒度通常是整篇文档。这种做法往往会返回过多的信息，用户很难从中快速找出自己所需的信息。从自动的方法来看，从一片长文本文档中定位答案并抽取的过程也较为复杂。在这样的情况下，味了减少后续模块中被处理的文本文档的大小，为了方便管理、准确定位，段落检索的概念开始为人们所重视（[Liu 2002], [Cui 2005]）。该模块首先从海量数据库中检索出可能包含答案的若干相关文档；然后，从这些文档中过滤掉与问题无关的段落；最后，通过计算剩余段落与问题的相关度，仅将少数高分段落返回，作为答案抽取模块的输入。与整篇文档检索的想法相比，段落检索具有以下优势：

1. 减少了所需要处理的文档片段的长度和规模，能够快速有效地进行语法分析；
2. 提高了文档相关性确定的准确性，避免了组成查询的关键词在文档中散布而造成的检索错误，也避免了文档长度对相关度的影响；
3. 提高了相关内容呈现的有效性：在特殊情况下，用户提出的问题需要大段文字的描述才能够解决，此时也可以直接将该模块所返回的段落作为答案。

[Tellex 2003]细致地考察了八种最好的段落检索算法。实验结果表明，基于密度的算法可以获得相对较好的效果。所谓基于密度的算法，就是通过考虑关键词在段落中出现次数和接近程度来决定这个段落的相关性。目前表现比较好的段落检索算法有三个：MultiText算法[Clarke 2000]、IBM算法[Ittycheriah 2000]和SiteQ算法 [Lee 2001]。以下方便对这三个段落检索算法进行介绍。

**MultiText算法**

MultiText算法将问题与文档都视为一个词序列。记问题的词序列为Q，文档的词序列为D。

考虑Q中出现的词的一个子集T。若D的一个连续子序列S中的词都是T中的词，则称S**满足**T。若S满足T且S的任何连续子序列S’都不满足T，则称S是T的一个**覆盖**。MultiText算法的基本思想是计算Q的所有子集在所有段落上的覆盖，并据此检索出与Q相关的段落。

对于一个词t，MultiText算法使用一种类似IDF思想的方式来计算其权重：

其中，指t在所有段落中出现的总次数，N则指整个语料库的总词数。

随后，一个词集T的权重被定义为其中的所有词的权重之和：

如果词序列是T的一个覆盖，则S和T之间的分值可以被定义为：

依据C(T,S)对所有可能的(T,S)对进行排序，并取出排在前列的若干个S。对于其中的每一个连续子序列S，MultiText算法选取其中间位置，并在原文档中向前向后各延伸100个单词，截取出一个总长度为200的段落，加入到段落检索结果列表中。

**IBM算法**

IBM算法从多方面特征来计算问题与文本段落的相关性。这些特征包括：

* 词匹配特征：即问题与段落共现的词的IDF值之和；（＋）
* 领域匹配特征：基于WordNet同义词典，若段落中有些词是问题中的某些词的同义词，则将这些词的IDF值进行加和；（＋）
* 词失配特征：即出现在问题中但未出现在答案中的词的IDF值之和；（－）
* 散布特征：将段落视为一个词序列，计算该词序列中最先匹配到问题中的词的位置与最后匹配到问题中的词的位置之间有多少个词无法匹配到问题中的词；（－）
* 词聚集特征：将段落视为一个词序列，统计其中有多少对相邻的词能够匹配到问题中的词。（＋）

其中，带（＋）标记的特征意为该特征的值越大，则IBM算法越趋于给这个段落一个较高的得分；反之，（－）标记意味着该特征的值越小，则IBM算法越趋于给这个段落一个较高的得分。

对这五个特征进行带符号的求和，即得到了问题与文档之间的相关性分值。

**SiteQ算法**

SiteQ算法将段落细分为句子。一个段落的得分被定义为该段落中所有句子的得分的总和。句子的得分是基于其中包含的问题中的词的稠密程度计算的。更具体地，一个句子的得分的计算公式为：

其中，意为问题中的第i个词的权重。在[Lee 2001]中，该权重取决于词性、WordNet等多方面因素，但在后续应用中，一般基于IDF值来设置此权重。即：如果该词在句子中出现，则其权重为IDF值；否则，权重为0。此外，指的是句子中与问题中的词能够匹配上的第j个词的权重；dist(j , j+1)指的是匹配到的第j+1个词与第j个词之间的距离；matched\_cnt指问题中有多少个词能够被该句子所匹配到；是一个超参数，用于调节词的匹配度（）与稠密度（）之间的权重。

综合来看，MultiText算法、IBM算法、SiteQ算法这三个算法虽然在设计和实现细节上有很大的差异，但是都使用了IDF值的总和，且都考虑了邻近关键词之间距离的因素。然而，基于密度的算法只考虑了独立的关键词及其位置信息，没有考虑关键词在问句中的先后顺序，也没有考虑语法和语义信息。针对这一问题，[Cui 2005]提出了一种基于模糊依赖关系匹配的算法。这种算法把问题和答案都解析称为语法树，并且从中得到词与词之间的依赖关系，然后通过依赖关系匹配的程度来进行排序。实验结果表明，这种方法的检索效果比基于密度的算法好。

传统检索方法一般只需要处理关键词，而问答系统需要处理更多的语法、语义信息。因此，部分问答系统也把语法、语义等信息添加到索引中，丰富了传统的索引，以提高检索效果。[Radev 2000]把一些关键词或者词组的属性放入索引，这样构造的查询包含关键词和答案的属性要求。例如，对于一个问时间的问题，构造的查询包含关键词和时间的属性，返回的段落中要求包含时间。[Bilotti 2007]把问题变成一个结构化的查询，表达查询词和段落中应该包含的某些词的属性。为了解决问题关键词的顺序问题，[Katz 2003]把句子解析为<主，谓，宾>三元组的形式，然后加入索引。另外， [Chu 2006]还索引了句子中词和词组的语义关系。

### 答案处理模块

段落索引模块输出的有序的段落集合被输入到答案处理模块中，以提炼答案。在答案处理模块中，系统需要对段落中的答案进行识别、抽取、验证和表达等一系列的处理。首先，利用相应的语法及语义分析规则，对每个输入的段落进行语义分析；而后，根据这些语义信息，以及问题处理模块的到的关于答案的信息推断出答案的准确位置，并将其抽取出来作为候选答案。为了检验答案的正确性，系统还将对候选答案进行评估以及排序，并将最可能正确的答案返回用户。我们将答案处理模块视为两个步骤的组合：

1. 候选答案集合的生成
2. 答案提取

#### 候选答案集合的生成

通过问题分析，已经获得问题的类型目前，问答系统的答案处理模块一般都是面向事实类型的问题的。对于非事实类型的问题，可以直接将段落检索的结果返回给用户作为答案。大多数事实类型的问题对应的答案比较短，可能是实体名，如人名、地点等；可能是抽象名词，如人类、学科、树木、植物等；也可能是数字，如距离、速度等。对于这类问题，可以通过找到相应类型的词、词组或者片段来回答。目前，自然语言处理领域命名实体的识别已经能够达到非常好的效果，如隐马尔可夫模型（hidden Markov model, HMM）活着条件随机场模型（conditional random field, CRF）。对于实体名词词表，除了利用WordNet等字典之外，还可以采用一些概念性名词和具体名词作为训练的种子，用bootstrap方法从文档集活着互联网中找到这种连接概念性名词和具体名词的模式，再根据这种模式提取更多的具体名词，多次迭代可以发现更多的<概念名词，具体名词>词对和相应的模式[Mann 2002]。另一种简单的方法是直接利用Web资源（如Wikipedia）中具体名词列表。对于抽象名词，通常是构建一个名词列表，若片段中含有这个列表中的词，就作为答案返回。对于数字度量，可以通过正则表达式来获取。例如，距离的一个模式是数字跟上距离单位，如5米。通过在文本中匹配相应问题类型的短语，就构成了候选答案集合。

#### 答案提取

与问题处理模块相类似，答案处理模块会使用相应的信息抽取策略。这些方法从语义描述能力的不同来分，包括：基于表层特征的答案提取，基于语法结构匹配的答案提取，基于模版的答案提取，基于统计模型的答案提取，等等。下面来具体介绍分析一下这些方法。

**基于表层特征的答案提取** 常用的表层特征是答案周围段落的一些特征，如段落和问题关键词的相关程度、问题关键词之间的距离、问题关键词和候选答案的距离等。一般来说，段落相关程度越高，问题关键词之间以及问题关键词和候选答案之间的距离越接近，则该候选答案越可能是问题的答案。另一个常用的特征是该候选答案出现的次数。对于一个比较大的文档集，一个问题的答案可能反复出现，出现的次数越多，则它越可能是正确答案[Lin 2003]。

**基于语法结构匹配的答案提取** 表层特征没有考虑语法、语义的因素，容易出错，特别是词相同，但词序不同的情况。[Light 2001]指出，这种基于实体识别和表层特征的方法的性能上限是70%。味了客服基于表层特征抽取答案的缺陷，基于语法结构匹配的答案提取方法被提出。其中一种方法是基于依存关系分析，将问题和候选段落转换成<主，谓，宾>三元组，删除句子中的修饰成分，就可以从文本三元组中获得答案而不产生混淆[Katz 2003]。另一种改进方法是建立问题到答案的逻辑表示[Moldovan 2002]。逻辑表示是介于句法解析表示和深层语义表示之间的一种表示形式，它可以通过对解析获得的句法树进行一些规则计算获得、表达主语、宾语、前置词、复杂的名词性短语、附属的形容词或副词之间的关系。

**基于模版的答案提取** 基于模版的答案提取方法通常被应用于将问题模版作为问题特征的问答系统中。在问题被映射到问题模版之后，可以直接获取该问题模版对应的答案模版，最后再利用答案模版从候选段落中抽取答案。味了能够保证答案的正确性，每个答案模版只能和一个问题模版相对应。该方法虽然不能分析候选段落的语义信息，但是通过这种模版间的一一对应关系，保证了所抽取到的信息的准确性。例如，TREC-10中的系统INSIGHT[Soubbotin 2001]通过人工分析的方法，将定义类问题的解答方式分为六种，并构建了六个答案模版，以从信息来源中抽取答案。利用这些答案模版，INSIGHT在该届TREC的比赛中赢得了冠军。淡然，模版的粒度和涵盖程度，对系统得到正确答案的成功率和答案质量都有决定性影响。于是，这种方法也带来了繁重的模版制定工作。为了解决这一问题，可以通过在训练数据上自动学习以得到模版。例如，[Cui 2004]提出了一种软模式的方法，来处理定义类问题答案的抽取。

**基于统计模型的答案提取** 在上述答案提取方法之外，研究人员还尝试用统计模型对答案提取进行建模。目前两个有代表性的建模方法为：一种是噪音信道模型，该模型把问题看成目标信息，把答案看成源信息，假设源信息需要通过一个包含噪音的信道，则转换概率可以通过一组训练数据（问题答案对集合）训练获得[Echihabi 2003]；另一种是用无向图模型对答案提取过程进行建模 [Ko 2007]。

## IBM Watson

前文介绍了问答系统的研究背景、基本概念与基本结构。本节将具体介绍IBM Watson，近年来最负盛名的开放领域问答系统。

Watson是由IBM公司研发的一个开放领域问答系统[Ferrucci 2012]。该系统在2011年2月的美国问答节目《Jeopardy!》上一战成名。在这次节目中，Watson战胜了这一节目的两位冠军选手，这被和1996年同样来自IBM的“深蓝”战胜国际象棋大师卡斯帕罗夫相提并论，被认为是人工智能历史上的一个里程碑。从技术角度来看，2011年参加“Jeopardy！”电视问答挑战赛时，Watson做了一件事——用自然语言进行深度问答。近年来，IBM公司已经对Watson在外延上进行了极大的扩展。如今，问答只是Watson具备的众多能力之一，截至2015年10月，Watson已经拥有包括问答在内的28项能力，并形成了一系列数字服务或API。在本文中，我们所说的Watson特指其中的自然语言问答功能。

我们可以通过2000次Jeopardy竞赛的数据来分析人类在回答这些问题时的表现。平均上看，一个人类选手会对40%到50%的问题给出答案，答案的正确率在85%到95%之间。为了击败人类选手，Watson的目标是：以至少85%的正确率回答70%以上的问题，且回答每个问题的反应时间只有数秒。

在Watson的处理流程中，共有数百个具体算法合作执行，以获得问题的答案。这些算法按照如图 3‑7所示的整体架构[Ferrucci 2012]进行组织。

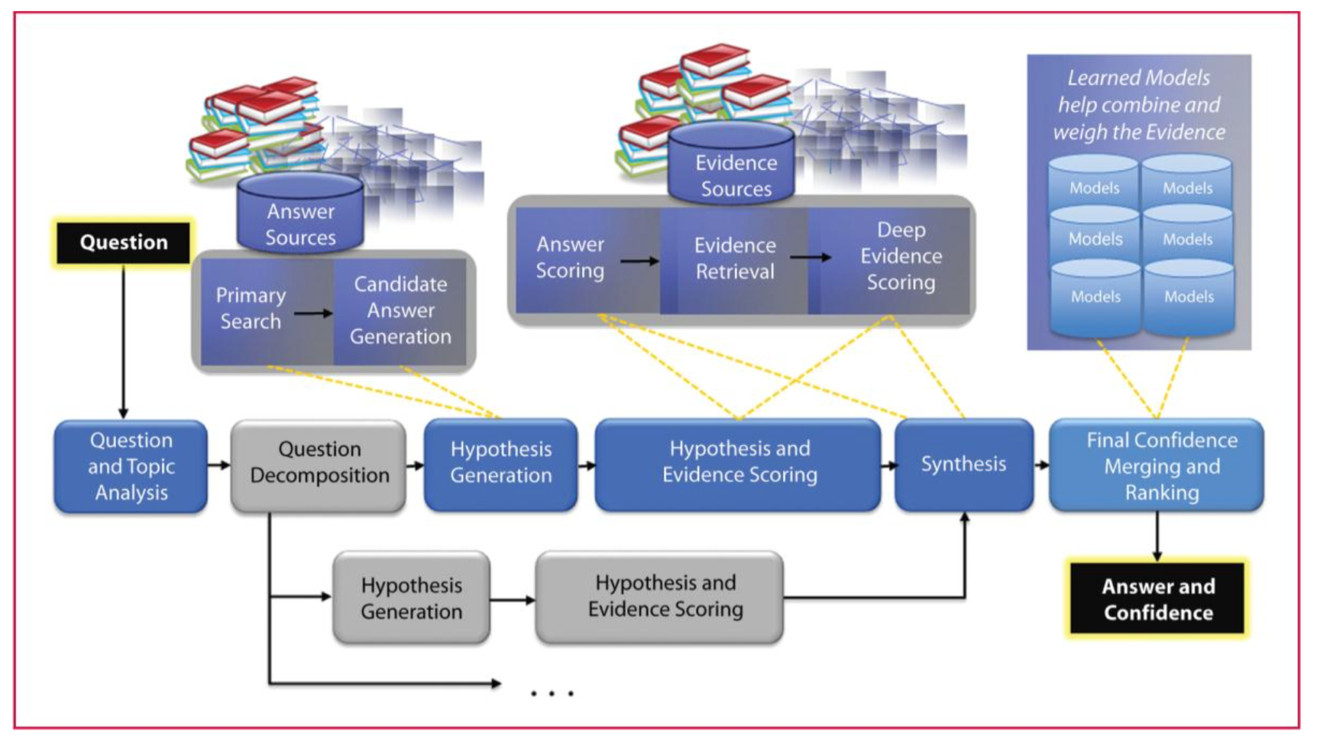


图 3‑7 IBM Watson的整体架构

本小节将按照如下顺序对Watson问答系统进行介绍：

1. 信息来源；
2. 问题分析；
3. 假设生成（即生成候选答案集合）；
4. 最终答案生成；
5. 其它技术。

### 信息来源

在信息来源方面，Watson主要做了三项处理：信息来源获取、信息来源转换、信息来源扩展[Chu 2012]。

**信息来源获取**，指的是获取能够对问题与答案有很好的覆盖度的数据集。Watson所采用的初始数据集为Wikipedia上的半结构化网页数据，一共包含超过3,500,000个页面，合计13GB。对于Jeopardy竞赛的历史问答集，这一数据集大约能够覆盖其中的77.1%的问题。在此基础上，Watson采用一种迭代式的失配分析方法来获取更多的信息来源，以覆盖更多的问题。在这一方法中，首先需要找出最经常发生失配的问题类型。这些问题类型包括：逆定义类问题（给出定义描述，要求答案给出相应的实体），引用类问题（可以是对引用中缺失的词/句子进行补充完整，也可以是要求给出引用内容的来源），圣经的细节，书籍和电影的情节等等。针对这些问题类型，Watson研究组搜集与之对应的数据源来覆盖它们，包括：维基词典、维基语录、各种不同版本的圣经、古滕堡项目中搜集的各种热门书籍，等等。通过这样的方式，Watson系统性地对主要失配的问题类型进行了信息来源补充。然而，还有一些特殊的问题并没有被覆盖到。为了覆盖这些问题，Watson课题组又加入了一些额外的开放领域数据集合，例如：辞典、新闻报道等等。

**信息来源转换**，指的是对信息来源中的原始数据进行一定的预处理，使其更符合后续的句子处理与答案抽取等工作的要求。图 3‑8和图 3‑9给出了一个简单的示例。图 3‑8是一段定义类型的语料，其中的句子大多没有主语（如“A set of structured activities.”），不利于后续处理。Watson将利用文档的标题，对这些句子进行重构。例如，对于前述的句子，转换后的句子是：“A program is a set of structured activities”。

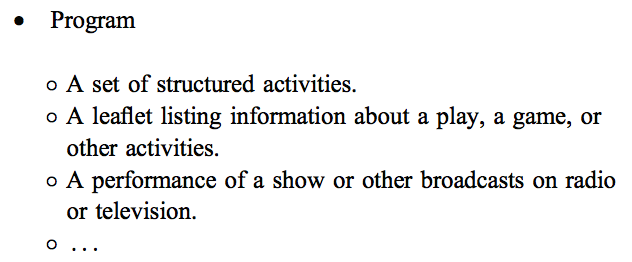


图 3‑8 定义类型的语料的一个示例

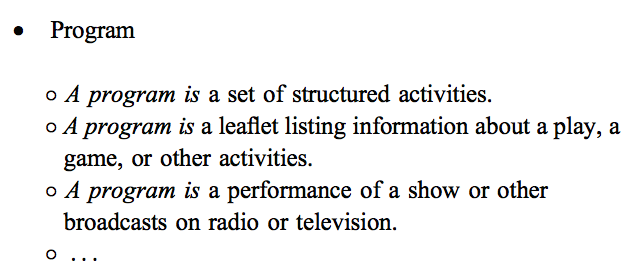


图 3‑9 对定义类型的语料的重构示例

**信息来源扩展**，针对的是辞典、字典等类型的信息来源。这些数据条目清晰，但这也意味着缺乏冗余。对于一个概念或实体，没有更多的数据从不同的侧面对其进行补充描述，这样的信息量很可能是不够的，导致有一些问题无法从中找到答案。因此，需要利用海量的互联网网页数据对已有的信息来源进行扩展，增加信息的冗余程度与丰富程度，以提高对问题的覆盖率。对于原信息来源中的一个文档，首先为其生成候选的扩展文档集。候选的扩展文档集一方面包含原文档中的链接所指向的互联网页面，另一方面包含以原文档中的关键词进行互联网搜索之后得到的高相关性网页。之后，对候选的扩展文档集进行段落切分与评分，选取得分较高的若干个段落，整合在一起作为原文档的扩展语料。图 3‑10给出了这一流程的示例。

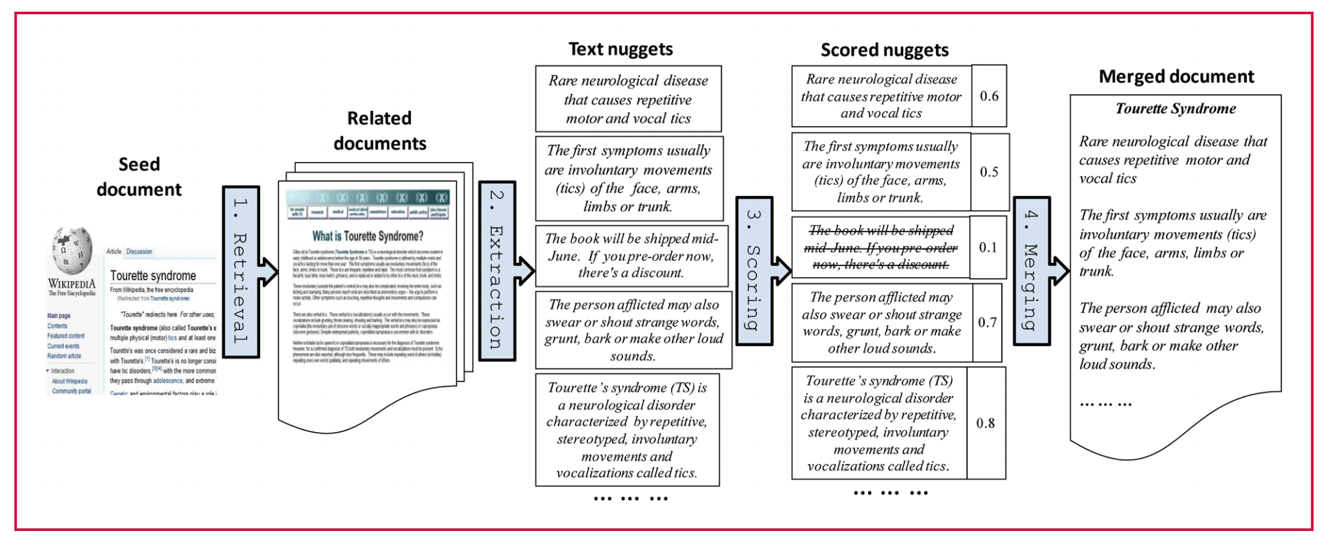


图 3‑10 信息来源扩展的流程示例

经过这三个步骤，Watson系统对Jeopardy的问题覆盖率从77.1%提升到了87.5%。

### 问题分析

通过问题分析，Watson从无结构的自然语言问题中识别出了句法与语义层面上的元素，并将它们编码为结构化的信息，从而在Watson的后续工作流程中进行使用。在Watson中，几乎所有组件都在某种程度上需要用到问题分析产生的结果。在Watson中，有许多检测规则和分类器被用于从问题中识别出各种特定的元素。不同的元素在后续处理中被不同的组件所使用。[Lally 2012]对Watson中的问题分析进行了概述。

在Watson的问题分析中，最重要的元素包括：问题焦点（focus）、词法答案类型（lexical answer types, LATs）、问题分类（Question Classification），以及问题部件（Question Section, QSection）。以下面这个Jeopardy问题为例：

POETS & POETRY: He was a bank clerk in the Yukon before he published “Songs of a Sourdough” in 1907.

问题的焦点指的是问题中指代答案的部分，在这个问题中，问题的焦点是“he”。在后续处理中，问题的焦点被用于与检索到的段落中的词进行对齐，从而抽取出答案。

LATs指的是问题中能够体现答案实体应该是什么类型的词的集合。在这个问题中，LATs为：“he”, “clerk”和“poet”。在后续处理中，LATs用于验证一个候选答案是否能用于回答这个问题。

问题分类指的是问题是否属于某个或多个较为宽泛的类型。本问题属于一个事实型的问题。对于Jeopardy的问题，还有其它的问题分类，包括：定义型问题、多选型问题、填空型问题、缩写型问题，等等。在后续处理中，问题分类的不同将导致Watson从不同的答案提取技术中选取最适合该问题类型的答案提取技术来进行处理。

问题部件指的是问题中需要特殊处理的片段，一般是一些约束。例如：“4-letter”或者“3-word”。在后续处理中，问题部件用于将问题分解为多个子问题的组合。

Watson实现问题分析的基本方法为：

首先解析问题句中的依存关系。Watson使用英文的槽文法（English Slot Grammar, ESG）解析器来实现依存关系的解析。

随后，基于这一解析结果，Watson的研究组用Prolog逻辑编程语言构造模版，从而可以通过模版规则匹配提取出问题中的各种元素。

### 假设生成

假设生成，指为一个问题生成候选答案集合。相比于最终给出的答案，假设生成所得到的候选答案集合并不十分看重候选答案的准确率，其重点在于尽量保证正确答案的召回率。当然，候选答案集合的规模也不宜过大，否则后续对各个候选答案进行评分的过程的代价将会过大，且容易被噪音所干扰。Watson的假设生成的概述可见[Chu 2012 a]。

在Watson中，假设生成主要由两部分构成：搜索和候选集合生成。

在搜索步骤中，Watson使用一种多因素的方法来获取与问题具有较高相关性的段落。相比于其它问答系统，Watson采用的段落检索策略在两方面进行了改进：首先，对于标题是概念或实体的问答，Watson采用了特殊的搜索策略以对标题与文本内容之间的关系加以利用；其次，Watson利用了从问题中提取出的句法和语义层面上的关系信息。

候选集合生成步骤中包含了多重候选集合生成策略，本文取其重的两个具有代表性的策略作为示例：

**基于文档标题的候选集合生成策略** 若搜索到的某个文档的标题是概念或者实体，则这个概念或实体就可以被作为一个候选答案。有的时候，文档标题具有消歧信息。例如，对于来自Wikipedia的文档，一个文档的标题是“Naomi(Bible)”，这说明该文档所述的Naomi是一个人；另一个文档的标题是“Naomi(band)”，这说明该文档所述的Naomi是一个乐队。Watson对这种情况进行了特殊处理，通过检查文档标题中的消歧信息是否与问题分析结果中的LATs相匹配来决定该概念或实体是否应该被加入到候选答案集合中。

**基于命名实体识别的候选集合生成策略** 从搜索到的文档中识别出若干命名实体，并将这些命名实体关联到Wikipedia中的条目上去。依照Wikipedia条目内的信息，可以知道该命名实体的类型。之后，比对该命名实体的类型与问题分析结果中的LATs是否匹配，来决定该命名实体是否应该被加入到候选答案集合中。由于Watson处理的最主要的文档是Wikipedia文档，而Wikipedia文档中的命名实体往往都带有超链接标签，因此这一策略中的命名实体识别主要指的是基于对超链接标签的解析找到对应的Wikipedia文档。

### 最终答案生成

在得到候选答案集合之后，Watson为每个候选答案生成证据，并根据证据是支持这一候选答案还是拒绝这一候选答案对证据进行打分。随后，这些证据的得分被综合起来，用于对候选答案进行排序，并给出答案的可信度。在Watson中，涉及到的证据多达上千种，人工设定这些证据的分值应该被如何综合起来是不现实的。针对这一问题，Watson使用机器学习算法，从已知的问题答案对中训练分值综合模型。[Gondek 2012]介绍了这一基于机器学习的答案综合与排序框架。

### 其它技术

## 参考文献

[Agichtein 2004] Agichtein, E., Lawrence, S., & Gravano, L. (2004). Learning to find answers to questions on the Web. *ACM Transactions on Internet Technology (TOIT)*, *4*(2), 129-162.

[Bilotti 2007] Bilotti, M. W., Ogilvie, P., Callan, J., & Nyberg, E. (2007, July). Structured retrieval for question answering. In Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval (pp. 351-358). ACM.

[Bouma 2002] Bouma, G., & Kloosterman, G. (2002). Querying Dependency Treebanks in XML. In *LREC*.

[Brin 1998] Brin, S., & Page, L. (1998). The anatomy of a large-scale hypertextual web search engine. Computer networks and ISDN systems, 30(1), 107-117.

[Carreras 2005] Carreras, X., & Màrquez, L. (2005, June). Introduction to the CoNLL-2005 shared task: Semantic role labeling. In Proceedings of the Ninth Conference on Computational Natural Language Learning (pp. 152-164). Association for Computational Linguistics.

[Chu 2006] Chu-Carroll, J., Prager, J., Czuba, K., Ferrucci, D., & Duboue, P. (2006, August). Semantic search via XML fragments: a high-precision approach to IR. In Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval (pp. 445-452). ACM.

[Chu 2012] Chu-Carroll, J., Fan, J., Schlaefer, N., & Zadrozny, W. (2012). Textual resource acquisition and engineering. IBM Journal of Research and Development, 56(3.4), 4-1.

[Chu 2012 a] Chu-Carroll, J., Fan, J., Boguraev, B. K., Carmel, D., Sheinwald, D., & Welty, C. (2012). Finding needles in the haystack: Search and candidate generation. IBM Journal of Research and Development, 56(3.4), 6-1.

[Clarke 2000] Clarke, C. L., Cormack, G. V., Kisman, D. I., & Lynam, T. R. (2000, November). Question Answering by Passage Selection (MultiText Experiments for TREC-9). In TREC.

[Collins 2004] Collins-Thompson, K., Callan, J., Terra, E., & Clarke, C. L. (2004, July). The effect of document retrieval quality on factoid question answering performance. In Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval (pp. 574-575). ACM.

[Cui 2004] Cui, H., Kan, M. Y., & Chua, T. S. (2004, May). Unsupervised learning of soft patterns for generating definitions from online news. In Proceedings of the 13th international conference on World Wide Web (pp. 90-99). ACM.

[Cui 2005] Cui, H., Sun, R., Li, K., Kan, M. Y., & Chua, T. S. (2005, August). Question answering passage retrieval using dependency relations. In Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval (pp. 400-407). ACM.

[Dang 2007] Dang, H. T., Kelly, D., & Lin, J. J. (2007, November). Overview of the TREC 2007 Question Answering Track. In Trec (Vol. 7, p. 63).

[Dave 2003] Dave, K., Lawrence, S., & Pennock, D. M. (2003, May). Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. In Proceedings of the 12th international conference on World Wide Web (pp. 519-528). ACM.

[Echihabi 2003] Echihabi, A., & Marcu, D. (2003, July). A noisy-channel approach to question answering. In Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1 (pp. 16-23). Association for Computational Linguistics.

[Ferrucci 2012] Ferrucci, D. A. (2012). Introduction to “this is watson”. IBM Journal of Research and Development, 56(3.4), 1-1.

[Gondek 2012] Gondek, D. C., Lally, A., Kalyanpur, A., Murdock, J. W., Duboué, P. A., Zhang, L., ... & Welty, C. (2012). A framework for merging and ranking of answers in DeepQA. IBM Journal of Research and Development, 56(3.4), 14-1.

[Green 1961] Green Jr, B. F., Wolf, A. K., Chomsky, C., & Laughery, K. (1961, May). Baseball: an automatic question-answerer. In Papers presented at the May 9-11, 1961, western joint IRE-AIEE-ACM computer conference (pp. 219-224). ACM.

[Hacioglu 2004] Hacioglu, K., Pradhan, S., Ward, W. H., Martin, J. H., & Jurafsky, D. (2004, May). Semantic Role Labeling by Tagging Syntactic Chunks. In CoNLL (pp. 110-113).

[Hao 2008] Hao, T., Hu, D., Wenyin, L., & Zeng, Q. (2008). Semantic patterns for user‐interactive question answering. Concurrency and Computation: Practice and Experience, 20(7), 783-799.

[Harabagiu 2000] Harabagiu, S. M., Moldovan, D. I., Paşca, M., Mihalcea, R., Surdeanu, M., Bunescu, R., ... & Morărescu, P. (2000). Falcon: Boosting knowledge for answer engines.

[Hu 2007] Hu, D., Li, H., Hao, T., Chen, E., & Wenyin, L. (2007, June). Heuristic learning of rules for information extraction from web documents. In Proceedings of the 2nd international conference on Scalable information systems (p. 61). ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

[Ittycheriah 2000] Ittycheriah, A., Franz, M., Zhu, W. J., Ratnaparkhi, A., & Mammone, R. J. (2000, November). IBM's Statistical Question Answering System. In TREC.

[Katz 2003] Katz, B., & Lin, J. (2003, April). Selectively using relations to improve precision in question answering. In Proceedings of the workshop on Natural Language Processing for Question Answering (EACL 2003) (pp. 43-50).

[Ko 2007] Ko, J., Nyberg, E., & Si, L. (2007, July). A probabilistic graphical model for joint answer ranking in question answering. In Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval (pp. 343-350). ACM.

[Kolomiyets 2011] Kolomiyets, O., & Moens, M. F. (2011). A survey on question answering technology from an information retrieval perspective. Information Sciences, 181(24), 5412-5434.

[Lally 2012] Lally, A., Prager, J. M., Mccord, M. C., Boguraev, B. K., Patwardhan, S., & Fan, J., et al. (2012). Question analysis: how watson reads a clue. Ibm Journal of Research & Development, 56(56), 2:1-2:14.

[Lampert 2004] Lampert, A. (2004). A quick introduction to question answering. Dated December.

[Lee 2001] Lee, G. G., Seo, J., Lee, S., Jung, H., Cho, B. H., Lee, C., ... & Kim, H. (2001, November). SiteQ: Engineering High Performance QA System Using Lexico-Semantic Pattern Matching and Shallow NLP. In TREC.

[Li 2002] Li, X., & Roth, D. (2002, August). Learning question classifiers. In Proceedings of the 19th international conference on Computational linguistics-Volume 1 (pp. 1-7). Association for Computational Linguistics.

[Light 2001] Light, M., Mann, G. S., Riloff, E., & Breck, E. (2001). Analyses for elucidating current question answering technology. Natural Language Engineering, 7(04), 325-342.

[Lin 2003] Lin, J., & Katz, B. (2003, November). Question answering from the web using knowledge annotation and knowledge mining techniques. In Proceedings of the twelfth international conference on Information and knowledge management (pp. 116-123). ACM.

[Liu 2002] Liu, X., & Croft, W. B. (2002, November). Passage retrieval based on language models. In Proceedings of the eleventh international conference on Information and knowledge management (pp. 375-382). ACM.

[Liu 2007] Xiaoli, L., Hu, D., Feng, M., & Wenyin, L. (2007, October). Semantic Pattern based Dependency Matching for Exact Answer Retrieval. In Semantics, Knowledge and Grid, Third International Conference on (pp. 262-265). IEEE.

[Mann 2002] Mann, G. S. (2002, September). Fine-grained proper noun ontologies for question answering. In Proceedings of the 2002 workshop on Building and using semantic networks-Volume 11 (pp. 1-7). Association for Computational Linguistics.

[Moens 2006] Moens, M. F. (2006). Information extraction: algorithms and prospects in a retrieval context (Vol. 21). Springer Science & Business Media.

[Moldovan 1999] Moldovan, D. I., Harabagiu, S. M., Paşca, M., Mihalcea, R., Goodrum, R. A., Gîrju, C. R., & Rus, V. (1999). Lasso: A tool for surfing the answer net.

[Moldovan 2000] Moldovan, D., Harabagiu, S., Pasca, M., Mihalcea, R., Girju, R., Goodrum, R., & Rus, V. (2000, October). The structure and performance of an open-domain question answering system. In Proceedings of the 38th Annual Meeting on Association for Computational Linguistics (pp. 563-570). Association for Computational Linguistics.

[Moldovan 2002] Moldovan, D. I., Harabagiu, S. M., Girju, R., Morarescu, P., Lacatusu, V. F., Novischi, A., ... & Bolohan, O. (2002, November). LCC Tools for Question Answering. In TREC.

[Moldovan 2003] Moldovan, D., Paşca, M., Harabagiu, S., & Surdeanu, M. (2003). Performance issues and error analysis in an open-domain question answering system. ACM Transactions on Information Systems (TOIS), 21(2), 133-154.

[Mollá 2007] Mollá, D., & Vicedo, J. L. (2007). Question answering in restricted domains: An overview. Computational Linguistics, 33(1), 41-61.

[Radev 2000] Radev, D. R., Prager, J., & Samn, V. (2000, April). Ranking suspected answers to natural language questions using predictive annotation. In Proceedings of the sixth conference on Applied natural language processing (pp. 150-157). Association for Computational Linguistics.

[Salton 1986] Salton, G., & McGill, M. J. (1986). Introduction to modern information retrieval.

[Sang 2005] Sang, E. T. K., Bouma, G., & de Rijke, M. (2005, July). Developing offline strategies for answering medical questions. In Proceedings of the AAAI-05 Workshop on Question Answering in Restricted Domains, Pittsburgh, PA, USA (pp. 41-45).

[Soubbotin 2001] Soubbotin, M. M., & Soubbotin, S. M. (2001, November). Patterns of Potential Answer Expressions as Clues to the Right Answers. In TREC.

[Tellex 2003] Tellex, S., Katz, B., Lin, J., Fernandes, A., & Marton, G. (2003, July). Quantitative evaluation of passage retrieval algorithms for question answering. In Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval (pp. 41-47). ACM.

[Woods 1977] Woods, W. A., & Kaplan, R. (1977). Lunar rocks in natural English: Explorations in natural language question answering. Linguistic structures processing, 5, 521-569.

[Zhang 2003] Zhang, D., & Lee, W. S. (2003, July). Question classification using support vector machines. In Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval (pp. 26-32). ACM.

[Zweigenbaum 2003] Zweigenbaum, P. (2003, April). Question answering in biomedicine. In Proceedings Workshop on Natural Language Processing for Question Answering, EACL (Vol. 2005, pp. 1-4).