基于本体推理的在线评测系统网络连接模型①

朱国进, 丁立波

(东华大学 计算机科学与技术学院, 上海 201620)

摘 要: 为了解决网络上不同系统之间连接的问题, 提出了网络连接模型. 针对网络连接模型中初始值无法由 HTML 解析器从页面中直接获取, 提出了基于本体推理的解决方案, 即通过对网页进行本体分析, 构建网页的本体模型, 然后在 KAON2 本体推理机中定义规则, 推理出网络连接模型的初始值. 实验以在线评测系统为例, 结果证明, 该方法具有很高的识别率, 大大提高了网络连接模型的自动化程度.

关键词: 本体; 本体推理; 本体构建; 网络连接; 在线评测系统

Network Connection Model for Online Judge System Based on Ontology Reasoning

ZHU Guo-Jin, DING Li-Bo

(School of Computer Science and Technology, Donghua University, Shanghai 201620, China)

Abstract: In order to solve the problem of the communication between different systems on the network, we put forward the model of the network connection. Aiming at the default value in network connection model that cannot be directly obtained by the HTML parser from the web page, this paper presents a solution based on ontology reasoning, which includes analyzing the web page with ontology, building ontology model for the page, defining rules in KAON2 ontology reasoning machine, and reasoning out the default value of network connection model. Experiments take an online judge system as an example. The results show that the method has high recognition rate and greatly improves the automation degree of the network connection model.

Key words: ontology; ontology reasoning; ontology building; network connection; online judge system

近年来, 随着 Internet 的迅猛发展, Web 已经成为全球传播与共享科研、教育、商业和社会信息等最重要和最具潜力的巨大信息源. 为了让这些资源得到最大化利用, 就迫切需要将网络上提供相似功能的一类系统连接起来, 在线评测系统[1](Online Judge System, 简称 OJ 系统)就是这么一类系统. 在平时的教学中, 教师为了阐述某个知识点, 往往需要从不同的 OJ 系统中选取相关的题目让学生练习.

现有的 OJ 系统大都基于 B/S 架构, 使用 HTTP 协议的 GET 或者 POST 方法进行通信. HTTP 协议是由键名和键值所组成的键值对(Key Value Pair)来完成通信的, 客户端必须首先知道用于通信的键名才能把它发送的数据(键值)有效地传送给服务器端. 在最近提出的面向知识框架的连接中[2], 键名是由人工从 OJ 网页中提取后预先配置在知识框架中来让连接系统读取的. 为了避免这种人工干预的局限性, 本文提出了一种从 OJ 网页中自动提取键名的方法, 其特点是自动地构建 OJ 网页的本体[3]实例, 并由 KAON2[4]推理机运行预定义的本体推理[5]规则来推导出 OJ 网页中对应各种通信的元素及其键名.

1 在线评测系统网络连接模型

在线评测系统网络连接模型位于应用系统与 OJ 系统中间, 充当代理角色, 见图 1, 其中 POJ、UVa 和 HDU 是目前国内外教学中影响很大的 3 个 OJ 系统.

① 基金项目:国家自然科学基金(60973121)

收稿时间:2014-03-17; 收到修改稿时间:2014-04-25

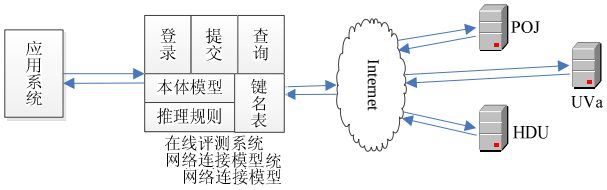

图 1　基于本体推理的网络连接方案

与面向知识框架的连接不同，本模型无需由人工预先设置键名. 例如，当收到应用系统请求以用户名“张三”和密码“ZS12”登录 POJ 系统时，本模型首先在键名表中查询对应的键名. 若查找失败，则启动如图 2 所示的处理流程: (1)由给定的 URL 读取 POJ 登录页面的 HTML 文件; (2)由本体模型构建该网页的本体实例; (3)由推理规则进行本体推理，确定该网页中的登录元素; (4)从登录元素中提取分别对应用户名和密码的键名“use_id1”和“password1”，并存入键名表. 一旦获得了键名，就构造 2 个键值对(user_id1，张三)和(password1, ZS12)发送给 POJ 系统，完成登录操作，最后返回登录结果给应用系统.

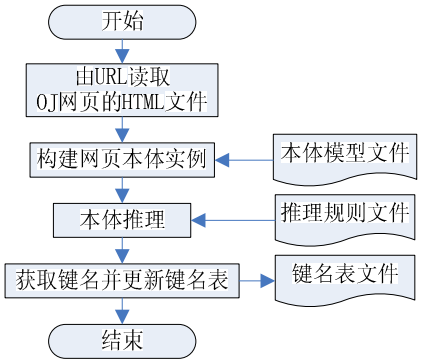

图 2　自动化获取键名流程图

2　登录网页的本体推理

2.1　机器读出的事实

我们以 POJ 系统登录页面为例，其 URL 为 http://poj.org/login，对应的 HTML 代码(部分)如下:

```
<form method=POST action=login>
User ID:<input type=text name=user_id1 ...>
Password: <input type=password name=password1 ...>
<input type=Submit value=login>
</form>
```

使用 HTML 解析器(如 HTML Parser, JSoup 等)，机器可以直接分辨出的事实是: 上述代码是一个由标签<form>标记的表单(记为 f)，并且表单 f 包含了分别被标签<input>标记的 3 个输入元素(记为 u、p、s)，它们分别具有的类别依次为 text、password 和 submit.

2.2　机器对事实的理解

这里所谓的机器对一组事实的理解，是指机器对这组事实在其内部产生的一种表示，依据这种表示机器能够使用逻辑规则[6]推导出正确的结果. 我们使用本体来表示这组事实，在上述的事实中有 3 个来自 HTML 中的概念，分别是表单、输入元素和输入类别. 在本体中引进这 3 个概念(分别记为 formCell、inputCell 和 inputType)，机器可以方便地在其内部产生出这些事实的本体表示. 例如，“f 是一个表单”这个事实可以由本体表示成: f 是概念 formCell 的一个实例. 机器可以把 f 作为概念 fromCell 的一个实例添加到本体中来在其内部形成对这个事实的本体表示.

为了表示“f 包含了 u”，还需要在本体中引进代表 fromCell 和 inputCell 这两个概念之间关系的“包含”(hasCell)这个属性. 此处，概念 formCell 被称为属性 hasCell 的定义域，而概念 inputCell 被称为属性 hasCell 的值域. 这样，机器就可以把有序对(f, u)作为属性 hasCell 的一个成员添加到本体中来在其内部形成对“f 包含了 u”的本体表示.

在 HTML 中，text、password 和 submit 分别用来表示输入元素的 3 个类别，我们可以把它们作为概念 inputType 的 3 个实例引进到本体中来，即事先直接把它们作为实例添加给本体中的概念 inputType，以表示它们是事先已知的事实. 在“u 是一个输入元素”这个事实中，u 是由机器在读 HTML 代码时即时产生的标识符，事先无法确定 u 所标识的对象，所以机器不能事先把 u 作为概念 inputCell 的一个实例添加到本体中去.

为了表示“u 的输入类别是 text”，同样也需要在本体中引进代表概念 inputCell 和概念 inputType 之间关系的属性“具有类别”(hasType)，其中属性 hasType 的定义域是概念 inputCell，而它的值域是概念 inputType. 这样，机器可以把有序对(u, text)是属性 inputType 的一个成员添加到本体中来在其内部形成对“u 的输入类别是 text”的本体表示.

至此，上述中所有事实，都可以用本体来表示.

2.3　登录表单的推理目标

根据上述事实，希望机器可以推导出如下结果:

(1)f 是一个用于用户登录的表单;

(2)u 是一个用于输入用户名的输入元素;

(3)p 是一个可以输入用户密码的输入元素.

为了达到上述推理目标, 首先需要使机器能够在其内部产生出对上述结果的本体表示, 因此还需要在本体中加入下列概念.

(1)概念 passwordCell, 它代表了所有可以输入用户密码的输入元素. 在 HTML 中, 只有其类别为 password 的输入元素是可以输入用户密码的. 因此, 概念 passwordCell 是概念 inputCell 的一个子概念.

(2)概念 loginForm, 它代表了所有的用于用户登录的表单, 它是概念 formCell 的一个子概念.

(3)概念 usernameCell, 它代表了所有的用于输入用户名的输入元素, 它是概念 inputCell 的一个子概念, 它的每一个实例都是某个登录表单中一个可以输入文本的输入元素.

需要指出的是, 与概念 passwordCell 是从 HTML 中引进的不同, loginForm 和 usernameCell 是应用领域中的两个概念, 它们并不存在于 HTML 中, 因而使用 HTML 解析器不能直接确定出它们的实例.

2.4 登录表单的本体模型

综上所述, 我们勾画了一个由 6 个概念、2 个属性和 3 个实例组成的登录表单本体模型, 记为 LFO. 使用本体五元组[7]来描述这个本体模型, Ontology = {C, R, F, A, I}, 其中 C、R、F、A、I 分别代表概念、关系、函数、公理和实例. 由于函数、公理在本文中涉及较少, 我们将其简化为三元组, 即 Ontology = {C, R, I}, 所以可以将登录表单本体模型描述为:

LFO = {Clfo, Rlfo, Ilfo}

Clfo = {inputType, inputCell, formCell, userCell, passwordCell, loginForm}

Rlfo = {

hasType(inputCell, inputType), hasIndividual(inputType, submit),

hasSubclass(inputCell, userCell), hasSubClass(formCell, loginForm),

hasIndividual(inputType, text), hasSubclass(inputCell, passwordCell),

hasIndividual(inputType, password), hasCell(formCell, inputCell)}

Ilfo = {text, password, submit}

我们得到的登录表单本体模型如图 3 所示.

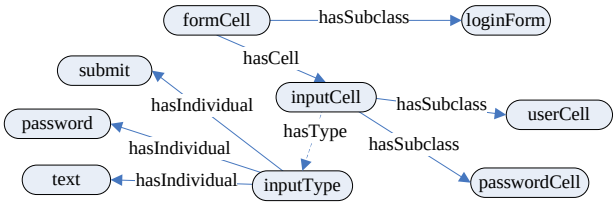

图 3 登录表单本体模型概念以及概念间关系图

2.5 登录表单的推理规则

我们使用 Prolog[8]逻辑编程语言来描述上述规则, 并使用 KAON2 推理机来执行 Prolog 推理规则.

规则一: 如果一个输入元素 P 具有类别 password, 那么 P 就是 passwordCell 的一个实例.

Prolog 逻辑语言描述如下:

classMember(P, passwordCell) :- hasType(P, password)

规则二: 如果一个表单 F 包含了三个输入元素 U、P 和 S, 并且它们的输入类别分别依次为 text、password 和 submit, 那么 F 就是 loginForm 的一个实例, U 就是 usernameCell 的一个实例.

Prolog 逻辑语言描述如下:

classMember(F, loginForm), classMember(U, usernameCell)

:- hasCell(F, U), hasCell(F, P), hasCell(F, S), hasType(U, text), hasType(P, password), hasType(S, submit)

注: 上面 P、U、P、S 在 KAON2 推理中为自定义的推理变量.

至此, 我们已经完成了一个带有推理规则的登录网页本体, 将其保存到文件中, 以便重复使用.

3 提交网页的本体推理

在 OJ 系统中, 提交网页是通过表单来获取用户提交的信息, 所以我们可以对表单包含的输入元素来识别提交网页.

提交表单(submitForm)在 HTML 源码中以表单形式存在, 所以它是概念 formCell 的一个子概念. 提交表单本体模型中包含以下概念: 题号域(idCell); 编程语言域(languageCell); 代码域(codeCell); 选择类型(selectType), 在 HTML 中由实例 select 和 radio 组成. 提交表单本体模型 SFO 三元组描述如下:

SFO ={ Csfo, Rsfo, Isfo}

Csfo = {formCell, inputCell, inputType, submitForm,

idCell, codeCell, languageCell, selectType}
Rsfo = { hasSubclass(inputCell, languageCell), hasType(inputCell, inputType), hasSubclass(formCell, submitForm),
hasCell(formCell, inputCell), hasSubclass(inputType, selectType),
hasSubclass(inputCell, codeCell), hasSubclass(inputCell, idCell),
hasIndividual(inputType, text), hasIndividual(inputType, password),
hasIndividual(inputType, textarea), hasIndividual(inputType, submit),
hasIndividual(selectType, raido), hasIndividual(selectType, select)}
Isfo = {text, textarea, submit, select, radio}

提交表单的推理规则是: 如果一个表单 F 包含了四个输入元素 C、P、B 和 L, 并且它们的输入类别分别依次为 textarea、text、submit 和概念 selectType 的一个实例 S, 那么 F 就是一个提交表单、C 就是一个代码域、L 就是一个编程语言域、P 就是一个题号域.

Prolog 逻辑语言描述如下:

classMember(F, submitForm), classMember(C, codeCell), classMember(L, languageCell), classMember(P, idCell)
:- hasCell(F, C), hasCell(F, L), hasCell(F, P), hasCell(F, B), hasType(C, textarea), hasType(P, text), hasType(L, S), hasType(B, submit),classMember(S, selectType)

4 查询网页的本体推理

OJ 系统的查询网页一般以表格的形式给出查询结果, 形成查询表格, 与上述两个表单本体模型不同, 查询表格本体模型需要获取列属性而不是键名.

查询表格(queryTable)是表格(formCell)的子概念, 由 HTML 中的<table>实现; 它的列(columnCell)由 HTML 的<th>或<td>实现, 其列类别(columnType)由所有可能的列标题组成. 不同类别的列各是 columnCell 的一个子概念, 而同一类别的列具有的类别是 columnType 的子概念. 如 columnCell 的一个子概念是评测结果列(resultCell), 后者所有可能的列标题 result、judge_status 和 verdict 所构成的一个类别 resultType 是 columnType 的一个子概念. columnCell 的其他子概念还有运行编号列(runCell)、题目编号列(problemCell)、提交时间列(subtimeCell)、编程语言列(languageCell)和运行时间列(runtimeCell), 它们具有的类别各是 columnType 的一个子概念, 依次分别为 runType、problemType、subtimeType、languageType 和 runtimeType. 查询表格本体模型 QTO 用三元组描述如下:

QTO = {Cqto, Rqto, Iqto}
Cqto={ tableCell, columnCell, columnType, queryTable, runCell, runType, problemCell, poblemType, languageCell, languageType, resultCell, resultType, runtimeCell, runtimeType, subtimeCell, subtimeType}
Rqto = {
hasType(columnCell, columnType)
hasCell(tableCell, columnCell)
hasSubclass(columnCell, resultCell)
… // 此处略去若干概念间继承关系
hasIndividual(resultType, result)
… // 此处略去若干概念与实例之间关系
}
Iqto = {run_id, problem, pro_id, result, judge_status, verdict, time, exe_time, run_time, language, submit_time, submission_time}

查询表格的推理规则是: 如果一个表格 T 包含了六个列 CR、CP、CL、CRR、CRT 和 CST, 并且它们的类别分别依次为一个运行编号类 TR、题目编号类 TP、编程语言类 TL、判题结果类 TRR、运行时间类 TRT、提交时间类 TST, 那么 T 就是一个查询表格、而 CR、CP、CL、CRR、CRT 和 CST 就分别依次为一个运行编号列、题目编号列、编程语言列、判题结果列、运行时间列和提交时间列.

以上规则用 Prolog 描述如下:

classMember(T, queryTable), classMember(CR, runCell), classMember(CP, problemCell), classMember(CL, languageCell), classMember(CRR, resultCell), classMember(CRT, runtimeCell), classMember(CST, subtimeCell)
:- hasCell(T, CR), hasCell(T, CP), hasCell(T, CP), hasCell(T, CL), hasCell(T, CRR), hasCell(T, CRT), hasCell(T, CST), hasType(CR, runType), hasType(CP, problemType), hasType(CL, languageType), hasType(CRR, resultType), hasType(CRT, runtimeType),

hasType(CST, subtimeType)

5 实验与应用

为了使本体推理机能够对给定的事实进行推理，需要把描述这些事实的信息添加到本体中. 以 POJ 登录页面为例，使用 JSoup 解析出该网页中的所有表单和输入元素以及它们之间的包含关系等等，然后将解析到的这些事实写入本体文件中，作为本体推理的数据输入. KAON2 本体推理机执行本体推理后，如果 f 出现在概念 loginForm 中，那么 f 就是一个用于用户登录的表单，否则 f 就不是一个用于用户登录的表单. 因此，需要对本体中的概念 loginForm 进行查询才能知道推理结果. 在 KAON2 中，执行语句 classMember(f, loginForm)可以查询 f 是否出现在 loginForm 概念中. 如果得到结果: http://poj.org/login#formCellOfForm0 is loginForm，则说明有一个登录表单 formCellOfForm0. 程序可以从该登录表单中提取出所需要的键名.

我们用 Java 语言实现了在线评测系统网络连接模型，并以 API 的形式为应用系统提供服务. 应用系统只需向该 API 提供题目来源 OJ、题目编号、编程语言以及编程代码，网络连接模型就会自动地完成登录、提交和查询等一系列操作，最后将评测结果、运行时间等信息返回给应用系统，整个过程无须人工参与. DHUVOJ(见 http://acm.dhu.edu.cn/onlinejudge)就是构建在该模型上的一个虚拟在线评测系统，该系统现已投入日常的程序设计课程教学中.

6 结语

本文提出了一种基于本体推理的方法获取网页中隐含的知识，该方法通过读入 HTML 网页中的一些事实，依据预定义的规则，判断网页的主题特征，而后从事实中自动化地获取我们需要的信息.

参考文献

1 Kurnia A, Lim A, Cheang B. Online judge. Computers & Education, 2001, 36(4): 299–315.

2 Zhu GJ, Chen YF. Knowledge-based links for automatic interaction with programming online judges. Journal of Software, 2013, 8(5): 1209–1218.

3 Gruber TR. A Translation approach to portable ontology specifications. Knowledge Acquisition, 1993, 5(2): 199–220.

4 Motik B, Studer R. KAON2–A scalable reasoning tool for the semantic web. Proc. of the 2nd European Semantic Web Conference (ESWC'05). Heraklion, Greece. 2005.

5 龚资.基于 OWL 描述的本体推理研究[硕士学位论文].长春: 吉林大学,2006.

6 王宗纬,朱国进,赵浪波,苏翔.基于 Ontology 和描述逻辑推理的 Web 题目资源检索.计算机工程,2006,32(18):225–227.

7 邓志鸿,唐世渭,张铭,杨冬青,陈捷.Ontology 研究综述.北京大学学报(自然科学版),2002,38(5):730–738.

8 Clocksin WF, Mellish CS. Programming in Prolog. 5th Edition. Springer. 2003. 140–165.