文章编号：2096-1472(2019)-10-01-05
DOI:10.19644/j.cnki.issn2096-1472.2019.10.001

用于知识库扩充的在线百科表格知识获取与融合

宋晓兆[1]，郑 新[2]，李直旭[1]，许佳捷[1]

(1.苏州大学计算机科学与技术学院，江苏 苏州 215006；
2.科大讯飞苏州研究院，江苏 苏州 215000)

摘 要：互联网中的HTML表格蕴含着丰富的结构化或半结构化知识，是知识库构建与扩充的重要数据资源。然而如何对HTML表格进行正确解析并获得三元组知识用于扩充知识库，则是一个很有挑战的问题。首先，HTML表格的结构各有不同。其次，表格与知识库中的实体和属性的表示不同，需要统一，即实体链接与属性对齐。本文首先提出了一个基于知识库的在线百科表格解析与知识融合框架，该框架可针对不同类别的表格进行知识抽取；并提出了基于知识库的表格实体链接和属性对齐方法，用以将表格中的知识与知识库进行匹配与融合。实验使用了126万在线百科表格数据为CN-DBpedia扩充约1000万三元组。

关键词：HTML表格；知识抽取；知识融合
中图分类号：TP391 **文献标识码**：A

Knowledge Extraction and Fusion over Online Encyclopedia Tables for Knowledge Base Augmentation

SONG Xiaozhao[1],ZHENG Xin[2],LI Zhixu[1],XU Jiajie[1]

(1.*School of Computer Science&Technology*,*Soochow University*,*Suzhou* 215006,*China*;
2.*Suzhou Institute*,*iFLYTEK*,*Suzhou* 215000,*China*)

Abstract:HTML tables in WWW have been flooded with (semi-)structured knowledge,which is an important source for knowledge base augmentation.However,it is a challenging problem to parse and extract triples in a correct way for knowledge base augmentation.Firstly,HTML tables have different types.Secondly,the descriptions of entities and attributes in different tables may be inconsistent with knowledge base,which needs to be matched and fused,i.e.,entity linking and property alignment.This paper first designs a table parse and knowledge fusion framework for the knowledge base,which is able to parse and extract knowledge in different types of tables.Additionally,an entity linking and property alignment method is proposed based on the knowledge base,to match and fuse the RDF triples with knowledge base.1.26 million tables in online encyclopedias are used in the experiment to augment 10 million triples for CN-DBpedia.

Keywords:HTML table;knowledge extraction;knowledge fusion

1 引言(Introduction)

迄今为止，所有基于在线百科构建的通用知识图谱[1-4]并未提出一种完全自动化的方法从在线百科的表格中挖掘知识，扩充知识库。现有的工作，如CN-DBpedia[4]，加入了端到端的深度学习模型从百科文本中挖掘知识，但是它并未挖掘百科表格知识。还有很多工作[5-9]致力于从整个互联网的表格中挖掘知识进行知识库的扩充，但是，他们仅仅使用单一类型的表格数据集[10,11]。比如这两个数据集ACSDb[10]和WDC Web Tables corpus[11]分别是英文和跨语言数据集，它们只含有关系表。关系表包含了多个实体(以行为单位)，一个实体有多个属性(以列为单位)。现有的表格数据集在类型不上并不完备，并且其中蕴含的知识可信度低。对此，我们研究了如何充分地使用在线百科表格扩充知识库。

2 问题分析(Framework overview)

使用百科表格扩充知识库面临的第一个挑战是表格类型的多样性问题。现有的工作[12]将互联网表格分成了10种类型，包括八种类型的知识表和两种类型的非知识表。而我们发现百科表格主要含有一种类型的非知识表和三种类型的知识表。其中，知识表如图1—图3所示，分为关系表、键值对表和枚举列表。目前最有效的表格分类方法[12,13]，选取表格特征信息和样本集来训练分类(识别)器。虽然它们能够精准地区别知识表和用于布局或导航的非知识表，但是在区分知识表的

基金项目：国家自然科学基金(No.61632016,61572336,61572335,61772356,61872258)；江苏省高校自然科学研究项目(No.17KJA520003,18KJA520010).

具体类型时，表现并不理想。为了精准地识别知识表的具体类型，需要构造相应表格的特征。我们发现百科表格中的属性与infobox的属性相似，属性集合与由知识库中属性构成的模式库有交集。因此，可以利用这个性质构造模式特征。与此同时，我们利用表格属性扩充模式库，更新特征值，这是一个迭代的过程。进而，我们提出了一种迭代扩充模式库、在线更新特征的表格识别器训练算法。

行政区	车牌代码	邮政编码	下辖行政区
济南	鲁A鲁S	250000	历下区、市中区、槐荫区、莱芜区、天桥区、历城区、长清区、章丘区、济阳区、钢城区、平阴县、商河县
青岛	鲁B鲁U	266000	市南区、市北区、李沧区、城阳区、崂山区、黄岛区、即墨区、胶州市、平度市、莱西市
淄博	鲁C	255000	张店区、淄川区、周村区、博山区、临淄区、桓台县、高青县、沂源县
枣庄	鲁D	277000	薛城区、市中区、峄城区、山亭区、台儿庄区、滕州市
东营	鲁E	257000	东营区、河口区、垦利区、广饶县、利津县
烟台	鲁F鲁Y	264000	莱山区、芝罘区、福山区、牟平区、龙口市、莱阳市、莱州市、蓬莱市、招远市、栖霞市、海阳市、长岛县
潍坊	鲁G鲁V	261000	奎文区、潍城区、寒亭区、坊子区、诸城市、青州市、寿光市、安丘市、昌邑市、高密市、临朐县、昌乐县
济宁	鲁H	272000	任城区、兖州区、邹城市、曲阜市、嘉祥县、汶上县、梁山县、微山县、鱼台县、金乡县、泗水县
泰安	鲁J	271000	泰山区、岱岳区、新泰市、肥城市、宁阳县、东平县
威海	鲁K	264200	环翠区、文登区、荣成市、乳山市
日照	鲁L	276800	东港区、岚山区、五莲县、莒县

图1 关系表

Fig.1 Relation table

CPU品牌	华为海思
CPU	巴龙5000+麒麟980
GPU	Mali-G72 MP10
运行内存	6GB
机身容量	512GB
电池类型	不可拆卸式电池
电池容量	4500mAh
传感器	重力感应,距离感应,光线感应,指纹识别,快速充电,陀螺仪,电子罗盘,OTG功能

图2 键值对表

Fig.2 Attribute/value table

山东省著名品牌							
海尔	张裕	齐鲁石化	青岛啤酒	海信	双星	雪柯玛	鲁能
得益乳业	鲁南制药	五征	新华制药	罗欣药业	新华医疗	万华	山东临工
皇明太阳能	兖矿	浪潮	中国重汽	孚日家纺	胜利油田	力诺	山东黄金
歌尔声学	谷神	中通客车	时风	山推股份	如意	太阳纸业	华勤
金锣	兰陵集团	沂州集团	华盛江泉	天元集团	国人西服	中科蓝天	泰山
今缘春酒业	希努尔 新郎	汇源					

图3 枚举列表

Fig.3 Enumeration table

使用百科表格扩充知识库面临的第二个挑战是如何将表格中抽取的知识与知识库中的知识融合的挑战。从键值对表中抽取知识时，每个三元组(即<s,p,o>)的主语(s)即所在百科页面的标题，通常一对一映射到知识库实体名称，不需要实体链接。谓语(p)和宾语(o)，对应表中每一行的键值对。比如图2中抽取的三元组<华为Mate 20，运行内存，6GB>。对于枚举列表，百科页面的标题作为主语，同样不需要实体链接。然而，关系表则需要进行实体链接和属性对齐。现有的将关系表与知识库匹配的算法框架TableToKnowledge,简称T2K[11]，采用迭代的方式进行实体链接与属性对齐。然而，它有两个不足：第一，T2K算法框架中并未考虑将表格内容整合[14,15]，它仅仅将单独的关系表与知识库进行匹配。然而，由于单一的表格实体数量少，属性稀疏，并且属性值常有缺失，这些表格不能直接与知识库匹配。于是，我们在T2K框架的基础上加入了整合表格内容的过程，提出了一个基于概念(本体)树的表格聚类算法。第二，T2K算法框架未采用有效方法生成实体链接候选集。它选择每个实体的候选实体集所属频率最高的概念，过滤不属于这些概念的候选实体。由此带来的后果是，长尾概念下的实体不能有效地进行实体链接，而这些实体对应的三元组往往是知识库所需要扩充的知识。于是，我们提出了基于“公共上位概念”的实体链接候选集生成方法。利用“公共上位概念”，我们不仅能够过滤无关概念下的实体，还能不遗漏长尾概念下的实体。

此外，本文针对在线百科表格数据集提出了一个知识融合策略。现有的互联网表格数据集体量大，热点知识出现次数多并形成偏态分布，通常以知识的交叠数量为特征训练知识融合模型。因此，同一条知识被抽取的次数越多，它的可信度越高。而百科表格中的知识分布均匀，有交叠的知识数量少，不能将交叠数作为特征。于是，我们提出了一种基于表格识别和实体链接准确率的融合策略。

综上，本文的主要贡献有：

我们提出了一种面向知识库扩充的在线百科表格知识获取与融合框架，可以一站式处理各类百科表格，抽取相关知识并融入知识库中。

为了对各种类型的表格进行对应的解析与处理，我们提出了一种表格识别算法。该算法可基于特征在线更新的表格识别器进行训练。

我们在T2K[11]算法框架的基础上增加了表格内容整合的过程，并利用“公共上位概念”生成实体链接候选集。

在本文的实验中，我们首先整合了百度百科和互动百科中126万个HTML表格，并将这些表格最终融入CN-DBpedia知识库中，实验表明本文的方法能够扩充约1000万三元组知识。

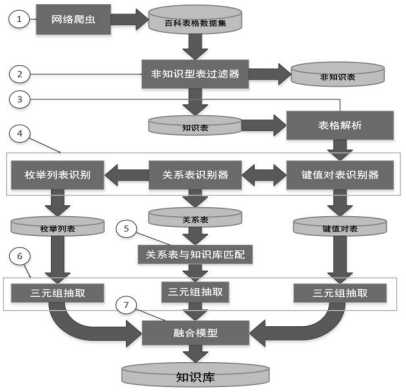

Fig.4 System framework

表1 表格类型定义

Tab.1 Table type taxonomy

名称	定义1—3
键值对表	固定列数为2，第一列为属性，第二列为属性值
关系表	第一行为属性名，至少存在一个主键，该列对应实体名称；第二行开始每一行是一个实体，除主键所在列外每一列表示一个属性
枚举列表	每个单元格，即行和列交叉的位置为实体名称，所有实体同属于一个概念,并与头实体(百科页面的标题)满足某个关系，如存在<caption>标签，则选择它作为每个三元组的谓语

3 框架概述(Key techniques analysis)

如图4所式，我们提出了一种用于知识库扩充的在线百科表格知识获取与融合框架，主要分为：

(1)网络爬虫：爬取不提供转储文件的在线百科，获取每个百科实体页面中的表格。由于百科表格的格式规范，以<table>标记的表格对象为主，因而百科表格数据集未考虑非<table>标记的表格对象。

(2)非知识表过滤：以<table>标记的表格对象分为带有知识的表格，和用于页面布局或导航的非知识表格，互联网中88%的HTML表不含有知识。借鉴互联网表格分类工作中的方法[12,13]，我们使用梯度提升树模型GBDT，作为非知识表过滤器。

(3)表格解析：将HTML格式的表格解析为csv格式，在内存中以二维数组的形式表示。同时在另外的数组中存储了单元格中的属性，如span和href。我们把带span属性，跨行跨列的单元格拆分。对于带href属性的单元格，我们使用其所指页面的标题作为链接实体。

(4)知识表类型识别：关系表和键值对的识别，采用了我们提出的基于模式特征在线更新的识别器训练算法，在识别的过程中，在线更新特征值，重新训练识别器。枚举列表的识别采取基于概念(分类)树辅助的启发式方法。

(5)关系表与知识库匹配：我们使用T2K[11]算法对关系表进行实体链接和属性对齐。此外，加入了我们提出的表格聚类算法，以及使用我们提出的“公共上位概念”进行候选集生成。

(6)三元组抽取：根据表1给出的三种表格的定义，按照相应的规则抽取知识。对于键值对表，所在百科页面的标题就是每个三元组的主语，表中每一行的键值对就是三元组的谓语和宾语。对于关系表，表格实体以行为单位，所链接的实体是每个三元组的主语，除主键所在列外每一列对齐到的属性是谓语，属性值是宾语。对于枚举列表，百科页面的标题就是每个三元组的主体，而表格中的每个实体名称则是每个关系三元组的宾语(尾实体)，尾实体链接采用与关系表实体链接相同的方法。谓语通过每个实体对在知识库中存在谓语的数量投票决定。

(7)融合模型：采用我们提出的针对百科表格数据集的融合策略。

4 关键技术分析(Key technical analysis)

在这一节，我们介绍了框架中三个关键技术的细节，它们是(1)知识表类型识别；(2)关系表与知识库匹配；(3)融合模型。

4.1 知识表类型识别

这一节中我们提出了识别三种表格类型的方法。表格中有两种类型的信息，属性信息和属性值信息。如果已知一些表格属性，那么我们可以利用它来识别表格的结构，从而能够帮助我们把属性对应到正确的属性值单元。由于表格是半结构化的数据，它的属性通常连续地出现在一整行或一整列。定位表格的属性会帮助我们识别表格正确的结构。对于键值对表和关系表，我们发现，表格属性与知识库中的属性有相同处，并且表格属性集合与由知识库属性构成的模式库存在交集。对此，我们将表格属性属于模式库的比例和个数作为模式得分特征，为键值对表和关系表分别训练了一个单层决策树，作为初始的表格识别器。在使用表格识别器识别表格后，将会含有一些不属于模式库的属性出现在表格中，但这些属性可能是其他表格的属性。于是，我们使用这些属性扩充模式库。模式库扩充后，训练集中表格的模式得分特征可能发生变化，需要更新，进而分类器模型又需要重新训练。如此往复，这是一个迭代的过程。如算法1所示，我们提出了基于特征在线更新的表格识别器训练算法。

算法1基于模式特征在线更新的识别器训练

输入：模式库$predict_{kg}$，知识表模式集合$Predict_{table}$，单层决策树*DStump*

输出：单层决策树*DStump*，扩充后的模式库$predict_{kg}$

1.*next_iteration=False*

2.for *predicttable* in $Predict_{table}$ do

3. computer $Score_{table}$

4. if *DStump*($Score_{table}$) *is True* then

5. if $predict_{table} - predict_{kg} \neq \varnothing$ then

6. $predict_{kg} \leftarrow predict_{kg} \cup (predict_{table} - predict_{kg})$

7. *next_iteration=True*

8. end if

9. end if

10.end for

11.if *next_iteration* is *True* then

12. update training set with new $Score_{table}$ and resume the training

13. if DStump performs better in testing set then

14. repeat 1 to 10

15 end if

16.else return *DStump* and $predict_{kg}$

17.end if

在算法1的输入中，模式库初始化为知识库中属性的集合；每个知识表的模式按行或按列获得(以属性表为例，它的模式由第一列中的每个属性构成)；识别器采用单层决策树模型，使用初始的得分特征进行训练。算法1的第3行计算了表格的两个模式得分，一个是属性属于模式库的比例，即$\frac{|predict_{table} \cap predict_{kg}|}{|predict_{table}|}$，另一个是属性属于模式库的个数。算法的第2行到第10行，计算每个未识别知识表的模式得分，如果有新的表被识别，则扩充模式库。每经过一轮迭代，都会重新训练一次识别器，原来的假负例在模式得分提高后会被识别为真正例，识别器的召回率会得到提升。当经过若干轮迭代后，模式库属性数量不再增加或识别器F1值不再提高时，我们将识别器和模式库返回。另外，可以在使用算法1完成弱学习器的训练后引入剩下的表格特征(如布局特征和内容特征)，通过boosting的方式训练一个更强的识别器。考虑到需要多次重复训练，于是我们选择单层决策树这样一个弱学习器作为识别模型，并且不引入其他特征。

在剩下未识别的知识表中，我们使用强规则识别枚举列表。我们把表格中每个单元格的内容假设为实体名称，通过知识库查找该实体名称对应的实体(实体和实体别称满足多对多的关系)，若每个实体别称都能映射到至少一个实体，则启发性地认为该表格为枚举列表。

4.2 关系表与知识库匹配

这一节，在T2K算法框架[11]中加入我们提出的基于概念(本体)树的表格聚类和基于公共上位概念的候选集生成方法。

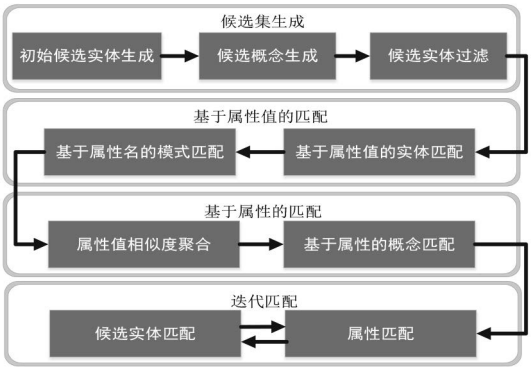

图5 T2K算法框架

Fig.5 T2K framework

4.2.1 T2K匹配框架

T2K算法框架将每个关系表视为一个小型关系型数据库，将关系表中的实体、属性和概念与知识库匹配。图5描述了T2K算法框架的主要步骤。它首先从知识库中获得候选实体，通过基于属性值的匹配得到候选实体的实体链接得分。然后以列为单位，选择属性值相似度的和最高的属性作为属性相似度，并计算这个属性对应的每个概念的得分，用得分最高的概念过滤候选实体。在过滤掉一些实体后，属性相似度发生了变化，需要重新选择，这是一个迭代的过程。T2K算法的先进之处在于，不同于传统数据库模式匹配，它在匹配过程中加入了概念(本体)的匹配，而概念(本体)是实体和属性的迭代匹配的桥梁。

4.2.2 基于概念的表格聚类

根据HTML表格与知识库匹配的经验，表格中实体数量越大，它们与知识库匹配的效果越理想。于是，在T2K框架的基础上将关系表内容整合。表格内容整合分为表格聚类和表格合并两个部分，前者采取了我们提出的算法2的方式，后者则是利用表格聚类的结果，将同类表格中相似属性合并到同一表格。这一节重点介绍了我们提出的基于概念的表格聚类算法。

此聚类算法以表格实体所属概念为特征，首先将每个表格表示为一个向量$T_i=[w_1,w_2,...,w_j]$，其中j是知识库中概念的数量。每一个维度对应的计算公式为：$w_j=\frac{|T_i.E\cap I(C_j)|}{|T_i.E|}$，其中，$T_i.E$表示表格$T_i$的实体名集合，$I(C_j)$表示知识图谱中概念$C_j$的实体名集合。接着，我们计算表格向量间的余弦相似度，然后采用如算法2所示的方法将表格聚类。

算法2基于概念(本体)树的表格聚类

输入：表格集合*Table*，相似度阈值*threshold*

输出：聚类簇*C*

```
1.Initialize clusters C=Ø
2.for table in Table do
3.    get vector T_i for table
4.    initialize flag f=False
5.    for cluster c in C do
6.        get vector T_c of the first table in c
7.        if cos(T_i,T_c) ≥ threshold then
8.            add table to c
9.            f=True
10.           break
11.       end if
12.   end for
13.   if f=False then
14.       initialize new cluster c={table}
15.       add c to C
16.   end if
17.end for
18.return C
```

可见，算法2是一种简单且有效的聚类算法，它的时间复杂度为$O(m\times n)$，其中n是表格的总数，m是聚类簇的数目，它远小于n。

4.2.3 基于公共上位概念的候选集生成

T2K[11]算法在候选集生成中，首先通过计算表格实体名与知识库实体名的相似度为每个实体生成*top k*个候选实体；然后为每个实体选择所属频率最高的概念，过滤不属于这些概念的初始候选实体。根据百科表格数据知识分布的特点，系统偏向于扩充长尾概念下的知识，使用高频概念不能有效地过滤初始候选实体。于是，我们提出了使用“公共上位概念”过滤初始候选实体的方法。

定义4(公共上位概念C_p)在概念树T中，如存在一个概念，其子概念构成的集合$C_{p.children}$与由每个候选实体集合E_i对应的概念集合C_i构成的集合C_{iset}都存在交集，我们把这个概念称为公共上位概念C_p，形式化为下列公式：

$$\{C_p\in T\mid\forall C_i\in C_{iset}:C_i\cap C_p.children\neq\varnothing\}$$

$$\text{s.t. } C_{iset}=\{C_i\mid\forall e\in E_i\cdot e.concept\in C_i\}$$

以图6为例，表格中存在中国、法国、日本三个实体名称，它们的候选实体集Ei分别为：{中华人民共和国}{法国(法兰西共和国)，法国(APA publications主编图书)}和{日本(日本国)，日本(山名)}，对应的概念集合C_i分别为分别为{东亚国家}{其他山脉，东亚国家}和{西欧国家，历史书籍}，则“国家”是这三个实体的公共上位概念C_p。而地形概念下的实体数量更多，它更可能成为高频概念。技术上，我们采用回溯算法遍历概念树得到C_p，过滤掉不属于C_p的候选实体。

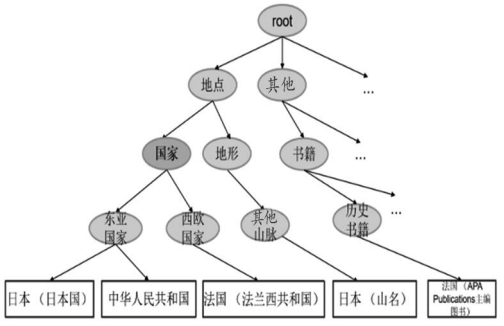

图6 一个公共上位概念的例子

Fig.6 Example of common superordinate concept

4.3 融合模型

由于互联网资源可信度较低，以往的工作在融合策略上

采用了基于知识库[21]或者网页排名的先验信任机制。而百科表格中的知识按领域分布均匀，属于长尾的较多，如果直接使用先验信任机制，那么这些长尾知识(知识库中的孤立节点)都不能被融合。考虑到百科资源具有很高的可信度，我们不需要采用先验信任机制，而应该以抽取器的准确率为指标，即识别器得分和实体链接相似度得分。我们将表格识别概率和实体链接相似度得分作为特征，为每种类型的表格分别训练一个逻辑回归模型。模型学习了两个特征的权重，以此得到知识的可信度。由于枚举列表和键值对表不需额外进行实体链接，他们的实体链接相似度得分均取1。

表2 表格识别结果评估

Tab.2 Table recognition assessment

内容	知识表			非知识表			关系表			键值对表			枚举列表		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
GBDT	0.983	0.972	0.977	0.962	0.977	0.969	0.717	0.720	0.719	0.767	0.764	0.766	0.523	0.549	0.536
算法1							0.992	0.802	0.887	0.989	0.762	0.861			
强规则													0.825	0.673	0.741

5 实验 (Experiment)

本文提出的方法已用于国内某个中文百科知识库的构建和扩充，采用的表格数据集来自百度百科和互动百科。由于百度百科与互动百科不提供转储文件，本文通过网络爬虫获得所有带<table>标签的HTML表格及对应页面信息。其中，互动百科的infobox信息同样采用<table>标签标记。在剔除互动百科340万个实体的infobox并过滤了15万个非知识表后，我们得到126万个中文百科知识表。同时，为了在公开数据集中验证实验有效性，本文使用中文百科格数据集扩充CN-DBpedia[4]，并且将实验结果与Ritze[11]的方法进行比较。

5.1 表格识别结果评估

非知识表过滤器模型20折交叉验证了5000个公开的已标注互联网表格和我们标注的1000个从中文在线百科中随机采样的表格，共获得126万知识表和15万非知识表。各类型表格识别器分别获得关系表34万，键值对表21万，枚举列表5万，剩下66万个表格属于复杂类型或难以融入知识库的表格。实验中，我们比较了使用算法1训练的单层梯度决策树表格识别器和未使用模式特征的梯度提升树识别器[12,13](记为GBDT)。我们的报告评估了准确率(P)、召回率(R)和F1值。表格识别评估结果如表2所示，实验证明使用我们的方法训练的识别器效果明显提升，尤其是准确率。

表3 表格实体链接算法比较

Tab.3 Table entity linking comparison

项目	P	R	F1
T2K	0.90	0.76	0.82
T2K+M	0.88	0.85	0.86
T2K+C	0.94	0.78	0.85
T2K+MC	0.92	0.84	0.88

表4 三元组抽取结果统计

Table 4 Knowledge extraction results

项目	关系表三元组	键值对表三元组	枚举列表三元组
百度百科	391万	168万	25万
互动百科	545万	223万	20万
总计	936万	391万	45万

5.2 关系表与知识库匹配结果评估

在4.2节中，我们在T2K[10]算法中加入了表格内容整合和利用公共上位概念的候选集生成的步骤。与Ritze[11]的工作不同的是，我们将中文百科表格与CN-DBpedia匹配。由于WDC Web Tables corpus[11]是来自全网的跨语言表格数据集，所以中文百科表格数据集可以认为是它的子集，同时CN-DBpedia也可以认为是跨语言知识库DBpedia的子集。在实验中，我们首先标注了100个关系表的实体链接结果，然后分别比较了这两个步骤对T2K算法中实体链接的影响。表3中T2K+M、T2K+C和T2K+MC表示分别加入表格内容整合，利用公共上位概念和综合利用两个步骤的T2K算法。如表3所示，表格内容整合使得更多的实体参与到与知识库的匹配过程中进而提升了召回率，而利用公共上位概念可以为实体选择语义相似度更高的候选实体集合，进而提升了准确率。

5.3 融合结果

如表4所示，我们挖掘出了近1400万的三元组知识。在最终入库时，除了要保证三元组的可靠度，还需要去重。我们采取了一个启发式方法，对于关系三元组(宾语是实体)，若头实体链接的知识库实体不含该三元组的尾实体，则直接入库。对于属性三元组(宾语不是实体)，若头实体链接的知识库实体的属性或属性值，与该三元组的属性，或属性值在相对编辑距离或基于字典的语义相似度上小于阈值则可以直接入库。去重后，我们可以向CN-DBpedia扩充约1000万三元组。

6 结论(Conclusion)

我们的工作提出了基于在线百科表格数据的知识库扩充框架，解决了表格识别和知识融合的挑战。实验结果证明了从百科中抽取的三元组的数量和质量能够用于知识库的扩充。

参考文献(References)

[1] Bollacker K.,Cook R.,Tufts P.Freebase:A Shared Database of Structured General Human Kowledge[C].AAAI Conference on Artificial Intelligence,Vancouver,2007(22-26):1962-1963.

[2] Mahdisoltani,F.,Biega,et al.YAGO3:a knowledge base from multilingual wikipedias[C].Proceedings of the Conference on Innovative Data Systems Research,Asilomar,2015:4-7.

[3] Lehmann,J.,Isele,R.,Jakob,et al.DBpedia-a large-scale,multilingual knowledge base extracted from Wikipedia[M].Semantic Web,2015,6(2):167-195.

[4] Xu B.,Xu Y.,Liang J.,et al.CN-DBpedia:A Never-Ending Chinese Knowledge Extraction System[J].In International Conference on Industrial,Engineering and Other Applications of Applied Intelligent Systems,Springer,Cham,2017:428-438.

[5] Boya Peng,Yejin Huh,Xiao Ling,et al.Improving Knowledge

Base Construction from Robust Infobox Extraction[J].NAACL-HLT,2019(2):138-148.
[6] Dong X.,Gabrilovich E.,Heitz G.,et al.Knowledge vault:a web-scale approach to probabilistic knowledge fusion[C].The International Conference on Knowledge Discovery and Data Mining,New York,2014:601-610.
[7] Ritze D.,Lehmberg O.,Oulabi Y.,et al.Profiling the Potential of Web Tables for Augmenting Cross-domain Knowledge Bases[C].International Conference on World Wide Web,2016.
[8] Zhihu Qian,Jiajie Xu,Kai Zheng,et al.Semantic-aware top-k spatial keyword queries[J].World Wide Web,2018,21(3):573-594.
[9] Venetis,P.,Halevy A.,Madhavan,J.,Pasca,M.,et al.Recovering semantics of tables on the web[J].Proceedings of the Vldb Endowment,2011,4(9):528-538.
[10] Cafarella M J.,Halevy A.,Wang D Z.,et al.WebTables:exploring the power of tables on the web[J].Proceedings of the Vldb Endowment,2008,1(1):538-549.
[11] Ritze D.,Lehmberg O.,Bizer C.Matching HTML Tables to DBpedia[C].International Conference on Web Intelligence,Mining and Semantics,Cyprus,2015.
[12] Crestan E.,Pantel P.Web-scale table census and classification[C].International Conference on Web Search & Data Mining.ACM,2011:545.
[13] Crestan E.,Pantel P.Web-scale knowledge extraction from semi-structured tables[J].ACM Press the 19th international conference Raleigh,North Carolina,USA Proceedings of the 19th international conference on World wide web,2010:1081.
[14] Yoshida M.,Torisaw K.,Tsujii J.A Method to Integrate Tables of the World Wide Web[C].In:Proceedings of the First International Workshop on Web Document Analysis,2001:31-34.
[15] Fan J.,Lu M.,Ooi B C.,et al.A hybrid machine-crowdsourcing system for matching web tables[C].2014 IEEE 30th International Conference on Data Engineering.IEEE Computer Society,2014.

作者简介：

宋晓兆(1995-)，男，硕士生.研究领域：自然语言处理，知识图谱.
郑 新(1990-)，男，硕士，工程师.研究领域：自然语言处理，知识图谱.
李直旭(1983-)，男，博士，副教授.研究领域：数据挖掘，知识图谱.
许佳捷(1983-),男，博士，副教授.研究领域：数据挖掘，时空数据库.

(上接第21页)

创建多个线程，这些线程就可以完成每一个逻辑业务请求，servlet程序始终保留在内存中，这样就可以非常迅速地响应客户端，缩短逻辑业务处理时间，提高系统处理时间。

(5)如果JSP文件被修改了，服务器就可以针对文件进行重新编译，将最新的编译结果保持在内存中，将最初的servlet程序覆盖掉，同时继续业务处理过程。JSP处理效率非常高，只需要在首次调用时进行转换和编译即可，这个过程中可能存在一些延迟，但是后期调用的时候就会加快处理速度。

4 结论(Conclusion)

JSP技术可以根据用户需求，开发IE浏览器端或智能移动设备端软件，并且适用于多种操作系统，比如Windows系统、Android系统、Linux系统的，提高了互联网应用软件的普适应和鲁棒性。JSP应用程序可以采用面向对象思想进行类和对象设计，为客户端和服务器端实现应用程序开发，有效整合互联网应用资源，进一步提高了分布式管理系统的开发便捷性和效率性。JSP技术不仅可以利用引擎执行终端应用程序和任务，而且不需要依赖服务器端的文件即可完成业务处理，建立一个良好的软件开发和处理机制。

参考文献(References)

[1] Yuan S,Chan,H.C.Stephen,Hu,Zhenquan.Implementing WebGL and HTML5 in Macromolecular Visualization and Modern Computer-Aided Drug Design[J].Trends in Biotechnology,2017,35(6):144-148.
[2] Yang T P,Beazley C,Montgomery S B,et al.Genevar:a database and Java application for the analysis and visualization of SNP-gene associations in eQTL studies[J].Bioinformatics,2010,26(19):2474-2476.
[3] Velden U V D,Abbas F,Armand S,et al.Java project on periodontal diseases.The natural development of periodontitis:risk factors,risk predictors and risk determinants[J].Journal of Clinical Periodontology,2010,33(8):540-548.
[4] 陈国华,詹宏昌,张文海,等.JSP技术及其在安全管理信息系统中的应用[J].中国安全科学学报,2013,13(1):45-47.
[5] 张波,张福炎.基于JSP技术的Web应用程序的开发[J].计算机应用研究,2011,18(5):99-101.
[6] 赵跃华,朱伟玲.基于SQLite数据库加密模块的设计与实现[J].计算机工程与设计,2018,29(16):4132-4134.

作者简介：

张明亮(1978-)，男，硕士，讲师.研究领域：计算机应用，信息研究.