一种基于主谓宾结构的文本检索算法

黄承慧[1,2] 印 鉴[1] 侯 昉[2]
(中山大学信息科学与技术学院 广州 510275)[1] (广东金融学院计算机系 广州 510520)[2]

摘 要 在文本检索领域,当前广泛应用的方法或者是考察检索词项与被检索文本的词频信息,或者是考察检索词项与被检索文本的语义相似性。这些方法忽略了检索词项与被检索文本的结构信息,检索结果有一定的局限性。通过分析检索词项与被检索文本句子结构的主谓宾信息,进而考察主谓宾结构中词汇的语义相似性,最终实现对文本的语义检索。实验表明,该方法能够有效提高检索的查准率。

关键词 文本信息检索,语义相似度,语法结构信息,检索算法
中图法分类号 TP311 **文献标识码** A

Improved Text Retrieve Algorithm Based on Subject-verb-object Structure

HUANG Cheng-hui[1,2] YIN Jian[1] HOU Fang[2]
(School of Information Science and Technology, SUN YAT-SEN University, Guangzhou 510275, China)[1]
(Department of Computer Science, Guangdong University of Finance, Guangzhou 510520, China)[2]

Abstract In text retrieve area, popular methods either considered word frequency or semantic information between retrieve terms and text corpus. These methods ignore the semantic structure information of retrieve terms and text corpus, and then the good result limits to some domains. This paper analyzed the subject-verb-object structure information of text, and then computed the similarity of the words where the words lie in the subject-verb-object structure, and finally implemented semantic information retrieving of texts. The experiment shows that the approach could improve the precision effectively.

Keywords Text information retrieve, Semantic similarity, Syntax structure information, Retroeve algorithm

1 引言

随着信息时代的到来,几乎所有传统的纸质文档都逐渐过渡为电子文档。这不仅因为电子文档易于保存,更重要的是电子文档更安全,使用更方便。任何一个组织机构都有一个包含大量电子文档的数据库。WWW的飞速发展则向全世界的人们呈现了一个海量的电子文档库。如何检索和应用这个庞大的文档库,是当前的一个热点话题。搜索引擎利用文本检索技术将用户搜索返回的结果根据文档的相似度进行排序,以便将更符合用户要求的结果排在前列。

文本检索的目标是识别与用户查询相关的文本,从而帮助用户从浩瀚的信息海洋中找到他们真正所需要的。因此,文本检索首要考虑的是定义什么是与用户查询"相关的"文本。在过去的研究中,许多不同的相关性定义导致了不同的检索模型被研究和验证。

尽管对文本的检索模型做了大量的研究,然而大多数检索模型或者基于检索词项的词频进行分析,或者考察检索词项与被检索文本中词项的语义相似性。这些研究忽略了用户查询与被检索文本中的语法结构信息,如主谓宾结构,导致了检索结果有着较大的局限,不尽如人意。

为了克服上述缺点,本文通过分析用户查询以及被检索文本中句子的主谓宾结构,针对主谓宾成分进行语义相似度计算,进而得到被检索文本与用户查询的相关性,实现了一种结合主谓宾结构的语义文本检索算法。实验结果表明,本文提出的方法能够有效地提高文本检索的查准率和查全率。

2 相关工作

目前主流的文本检索模型根据"相关性"的不同定义大致可以分为两类,最常见的是建立一个词频向量模型,用户查询和被检索文本之间的"相关性"由向量之间的相似性来衡量;其次是考察用户查询和被检索文本之间词项的相似性。

在基于"词频向量"的文本检索模型中,文本和用户查询被建模为高维词项空间的两个词项向量,每个词项赋予特定权重来反映该词项在文本或者查询中的重要性[1],词项的权重都基于词项的词频信息。与该查询"相关的"文本就由查询与文本两个词项向量之间的相似度来衡量。通常词项向量之

到稿日期:2009-10-26 返修日期:2010-01-14 本文受国家自然科学基金(60573097,60773198,60703111),广东省自然科学基金(05200302,06104916),广州市科技计划项目(2007Z3-D3071),高等学校博士学科点专项科研基金(20050558017),新世纪优秀人才支持计划(NCET-06-0727)资助。

黄承慧(1976—),男,博士生,讲师,主要研究方向为数据挖掘、语义Web、软件工程,E-mail:hch.gduf@163.com;印 鉴(1968—),男,博士,教授,博士生导师,主要研究方向为数据挖掘、人工智能;侯 昉(1975—),男,讲师,主要研究方向为计算机网络、语义Web。

间的相似度由余弦夹角或者 Jaccard 系数来定义。这种词项向量通常都具有高维、稀疏的特征，从而难以处理。为了克服这些问题，潜语义索引方法利用奇异值分解方法可以有效地将词项向量降维到可处理的范围[2]，进而对降维的词项向量进行处理，取得了较好的效果。

针对词频信息建模的词项向量模型虽然能够取得较好的效果，但是这些方法都忽略了词项的含义，因此有着一定的局限性。基于上述观察，人们开始研究词与词之间的相似度，以此改进检索的效率。考察词与词之间相似度的方法是使用一个外部的词典来构建词项的词义网络，通过考察不同词项在语义网络中的关系来决定词项之间的相似性。最常用的是普林斯顿大学研究开发的 WORDNET[4]。词与词之间相似度的研究为计算机理解文本提供了一个有效的途径，从而更好地帮助人们对电子文本进行自动化处理。

文献[3]利用 WORDNET 提供的同义词对用户的查询进行词项扩展，同时优化用户提交查询语句中的关键词组，以此提高检索效果。文献[5]利用潜语义索引技术和 WORDNET 作为本体，提出了一种独立于 WORDNET 知识但依赖于潜语义索引知识的模型，利用该模型扩展查询，取得了较好的效果。类似地，文献[6]则集成了 WORDNET、潜语义分析以及和领域相关的受控词汇等技术来改进检索结果的有效性。

为了更为精确地捕捉用户查询和被检索文本的相似性，近年来，越来越多的文献关注句子的相似性，并将其用于文本检索领域。文献[7]提出了一种结合语料库和知识的文本相似度量方法，该方法通过分析两个句子中包含单词的语义相似度，利用单词包含的信息内容(Information Content)进行加权，得到两个句子的语义相似度。文献[8]则考察了句子之间不相似的度量，利用该度量对句子进行聚类，进而自动生成文本的摘要信息。文献[9]分析了句子的动词结构，结合传统的 TF-IDF 技术，将文档句子结构中不同的、重要的概念赋以不同的权重，以此改进 TF-IDF 技术对于出现相同次数的词有着相同重要性的缺陷。

上述方法对句子的结构信息都未能做出深入的分析，在检索过程中忽略了文本的语法结构信息，不能精确地捕捉用户查询和被检索文本的相似性。

3 算法描述

TF-IDF 技术是当前主流的检索技术。该方法将文本看作是一个容纳词项的袋子，不考虑词项出现的顺序，也不考虑词项的含义。文本特征向量由文本中出现的词项在单个文本中出现的频率以及该词项在整个文本集中出现的频率来表示。每一篇文本建模为文中出现的 n 个加权词项组成的向量。该方法基于以下经验观察：

(1) 词频(Term Frequency)：某个词项在一个文本中出现的次数越多，它与文本的主题越相关；要注意在特定的语言环境下都有许多特定的词不具备这种特性而应将其排除，如中文的“的”、“地”，英文的“a”、“an”。

(2) 逆文本频率(Inverse Document Frequency)：某个词项在文本集合的多篇文本中出现越多，该词项的区分能力越差。例如在一个包含 1000 篇文本的集合中，如果某个词项 A 在 100 篇文本中都出现，而另一个词项 B 只在 10 篇文本中出现，则词项 B 比 A 具有更好的区分能力。

通过对文本集合中的每一个词项都进行上述分析，得到每一篇文本中每一个词项的 TF-IDF 值。之后利用这些 TF-IDF 值为每一篇文本建立一个向量模型，通过计算向量间的余弦相似度或者 Jaccard 系数来表示文本之间的相似性。

尽管这种简单的方法在实践中证明有较好的效果，然而随着互联网的发展，海量的文本数据要求我们能够更为精确地捕捉和刻画文本的含义，而不仅仅是文本词项出现的频率。例如一篇关于银行(bank)的文章和一篇关于河岸(bank)的文章，由于银行和河岸两者的词项都是 bank，基于词频的检索方法就很可能将它们都返回给用户。而一篇关于苹果和一篇关于橘子的文章(apple 和 orange)则可能因为用户检索水果一词而无法被检索。

利用词与词之间的相似度，改进的语义相似机制在词频向量的基础上，进一步捕捉了文本的含义，提高了计算机对文本的理解能力和检索效率。通过分析用户查询和被检索文本中词汇的语义相似性，以此决定一个给定的文档和一个用户的查询到底在多大程度上是语义相关的。最后根据检索文档与用户查询之间的相似度值高低进行排序，将检索结果以列表形式返回给用户。

尽管改进的语义相似性方法提高了检索精度，然而由于这种方法忽略了文本的结构信息，使得检索结果仍然存在不足。例如，有以下两个句子，“A tourist who comes from Beijing visit GuangZhou NanYue King Museum.”和“Beijing history museum is a famous and contains a lot of things that reveals the nature of Chinese culture.”，用户输入查询“Beijing Museum”，其本义是希望搜索北京的博物馆，基于词频分析和语义分析的方法都会检索到上述两篇文章；而如果是基于句子结构进行检索，则只会检索到后一篇文章。因此，如果能够根据句子结构进行检索，检索的精确度将会有进一步的提高。

本文利用自然语言处理技术，进一步深入分析了检索文本中句子的主谓宾结构，利用 WordNet 探讨两个句子间主谓宾成分之间的语义相似性，从而精确地捕捉了用户查询和被检索文本之间的语义相关性，更好地实现了用户检索的目标。

3.1 基于主谓宾结构的文本语义检索流程

本文提出的基于主谓宾结构的文本语义检索算法如下。

算法 1 基于主谓宾结构的文本语义检索算法

1)文本预处理：

a)将文件分析为一个个的句子，将从句也列为单独的句子。

b)进行命名实体分析，将句子改写为包含命名实体的句子。如 Mr Smith is playing basketball.→Person is playing basketball.

c)进行词源分析，不考虑动词的时态，还原动词。如 Person is playing basketball→Person play basketball.

d)利用终止词列表，删除句子中的 the,of,his 等词汇。

2)分析句子的主谓宾结构，将每一个句子的主谓宾结果用一个三元组的数据结构保存起来。对于没有宾语的不及物动词结构，宾语置空；有双宾语的动词结构，将构成两个宾语的词项放在一起。每篇文章则是由所有句子的主谓宾三元组组成的向量集合。

3)根据用户查询，构造适当的主谓宾三元组向量，对文本的主谓宾三元组向量集合进行检索。

4)根据检索结果与用户查询之间的相关性对检索结果大小排序。

上述算法中的关键是步骤 2)和步骤 3)，即设计主谓宾的三元组结构，以及根据主谓宾三元组对文本进行语义检索。

下面给出基于主谓宾结构的三元组数据结构描述，以及两个句子间基于主谓宾结构的相似度定义。

3.2 文本句子及句子相似度的定义

定义 1(句子) 句子 $ST=\langle s,v,o\rangle$ 是一个三元组，其中 s，v，o 分别为主语、谓语和宾语。主谓宾三者均由数量不定的单词所构成，同时还包含了该单词的语法性质，即属于名词、动词、形容词还是副词，分别用 n、v、a、ad 表示。例如 Person play basketball 的三元组表示为〈Person_n，play_v，basketball_n〉。

定义 2(句子的相似度) 两个句子 $ST_1=\langle s_1,v_1,o_1\rangle$，$ST_2=\langle s_2,v_2,o_2\rangle$，其中 s_1，s_2 为主语，v_1，v_2 为谓语，o_1，o_2 为宾语。考虑到主谓宾在句子中所起的作用不同，赋予三者不同的权值，则句子 ST_1 和 ST_2 的相似度 $\mathrm{Sim}(ST_1,ST_2)$ 由句子主谓宾的相似度加权得到。

$$\mathrm{Sim}(ST_1,ST_2)=w_1*\mathrm{Sim}(s_1,s_2)+w_2*\mathrm{Sim}(v_1,v_2)+w_3*\mathrm{Sim}(o_1,o_2) \quad (1)$$

式中，w_1，w_2，w_3 分别表示主语、谓语和宾语的权重，满足 $w_1+w_2+w_3=1$。为了体现实验的无监督特征，在本文的实验设置中，主谓宾 3 个权重相同，均为 1/3。

$\mathrm{Sim}(s_1,s_2)$，$\mathrm{Sim}(v_1,v_2)$ 和 $\mathrm{Sim}(o_1,o_2)$ 分别表示两个句子中主语、谓语和宾语之间的相似度。主谓宾之间的相似度，利用三元组向量中构成主谓宾的词项信息和语义信息进行相似度的计算。由于主谓宾可能由多个词项构成，因此主谓宾的相似度由组成主谓宾的多个词项相似度来表示。

例如，句子 ST_1：My hobbies are dancing and painting. 和 ST_2：I got the prize in drawing competition. 经过终止词删除和主谓宾分析后得到的三元组为

〈(hobby_n)，(null_v)，(dance_n，paint_n)〉

〈(null_n)，(get_v)，(prize_n，drawing_n，competition_n)〉

两个句子的宾语部分都包含多个词汇，则宾语部分(dance_n，paint_n)和(prize_n，drawing_n，competition_n) 的相似度计算如下。

首先从单词 dance 开始，计算 dance 与 prize，drawing，competition 之间的相似度，取三者的最大值。然后从单词 paint 出发计算 paint 与 prize，drawing，competition 之间的相似度，取三者的最大值。计算最大相似度的算术平均值作为句子 ST_1 与 ST_2 的相似度 $\mathrm{Sim}(ST_1,ST_2)$。

再从 prize 出发，计算 prize 与 dance，paint 的相似度，取两者的最大值；依此类推，求得另外两个单词与句子 ST_1 中最相似的单词相似度。计算最大相似度的算术平均值作为句子 ST_2 与 ST_1 的相似度 $\mathrm{Sim}(ST_2,ST_1)$。

最后计算 $\mathrm{Sim}(ST_1,ST_2)$ 和 $\mathrm{Sim}(ST_2,ST_1)$ 的平均值，作为句子 ST_1 和 ST_2 的相似度。

尽管通过删除终止词减少了用于进行相似度计算的词项，然而在文本集合中还是存在着许多不重要的词汇，它们的存在导致相似度计算的时间增加，同时影响了句子间的相似度。因为过多不重要的词项相似度，使得主谓宾相似度的加权平均值较低。因此，有必要过滤进行词项相似度计算的词项，确保参与相似度计算的词项有足够的表征意义。本文采用 TF-IDF 方法对文本计算词项的 TF-IDF 值，选取每个文本中 TF-IDF 值较大的词项，之后再进行主谓宾相似度的计算。

算法 2 利用 TF-IDF 方法过滤不重要词项

1)计算各文本词项的 TF-IDF 值。

2)将单个文本中的所有词项按 TF-IDF 值从大到小排序，同时累加所有词项的 TF-IDF 值。

3)选取特定比例 $\mu(0<\mu<1)$ 的 TF-IDF 值，过滤不重要的词项。根据信息论，具有较大 TF-IDF 值的词项包含了较多的信息内容，具备较好的区分文本的能力。反之，具有较小 TF-IDF 值的词项，则包含了较少的信息，因而难以区分文本。

4)将那些 TF-IDF 值在特定比例阈值 μ 以下的词项从主谓宾三元组中删除，以减少主谓宾相似度计算时对不重要词项的处理时间。

3.3 查询与文本的相似度定义

得到查询与句子之间的相似度之后，可以计算查询与文本之间的相似度，进而可以根据用户查询与文本之间的相似度对检索结果进行排序。本文采用查询与文档句子的算术平均值作为查询与文本之间的相似度，计算公式如下

$$\mathrm{Sim}(Q,D_i)=\frac{\sum_{j=1}^{n}\mathrm{Sim}(Q,ST_{ij})}{n} \quad (2)$$

式中，Q 表示用户查询，D_i 表示一篇文档，ST_{ij} 是属于文档 D_i 的句子。

用于计算词项之间相似度的方法有许多。本文采用 WordNet::Similarity[10] 工具包计算词项之间的相似度，该工具包实现了 8 种主流的词项之间相似度计算的方法。根据文献[11]，基于信息内容度量的相似度方法优于其它方法。因此，本文采用文献[12]所实现的词项相似度计算方法。

4 实验

实验数据采用业界广泛采用的 Ruters-21578 数据集(Re)。利用开源工具 Lingpipe[13] 对文本进行主谓宾结构分析后，有 10147 个文件具备主谓宾结构，从中剔出单词数目较少的文件，最终选取了 7875 个文件作为测试数据。

本文提出的句子相似度检索定义，需要根据词项的相似度来计算句子主谓宾结构之间的相似度。需要指出的是，计算两个词项相似度的 WordNet::Similarity 工具包是采用 Perl 语言编写的，计算效率较低。因此，本文对实验数据集中的词项相似度预先进行了计算，以名称将两个词项字符串连接在一起，中间以"##"分隔的形式，值为词项相似度的哈希表的形式保存在一个文件中。实际进行检索时，只需将该文件的内容读入内存构建一个哈希表，再从中查找已经计算好的相似度，从而减少算法的运算时间。

为了验证本文提出的方法是否能够比传统的检索方法效率高，我们在 Lucene 上建立了上述文本集合的索引，利用 Lucene 进行传统的检索实验，以及在不同的句子相似度阈值情况下进行检索。两者实验的对比结果如图 1 所示。图中横坐标 P@5，P@10，P@20，P@50 分别表示检索结果列表前 5，10，20，50 项时对应的精确度，纵坐标表示检索精确度。不同的数据系列 $\mu=0.6$，$\mu=0.8$，$\mu=1.0$ 等分别表示在选择不同比例的 TF-IDF 重要词项下所取得的实验结果。

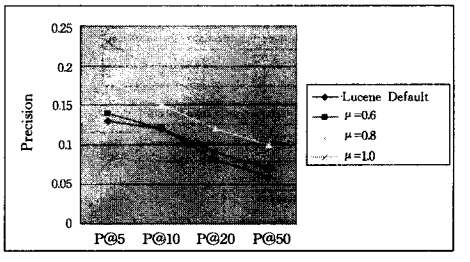

图 1 选取不同比例的重要词项和 Lucene 检索的对比实验

实验的总体结果是比较满意的。从图 1 可以看到在 Re 数据集上，对比传统的 Lucene 检索方法，即使选取 $\mu=0.5$ 时的 TF-IDF 重要词项，实验结果仍然优于传统的 Lucene 检索方法，说明在传统词频分析的基础上结合句子结构信息的方法是可行的。

结束语 本文针对传统的基于词频分析的检索方法增加句子结构的语义相似信息，并利用基于信息理论的词项语义相似度理论来计算句子之间的语义相似性，之后根据句子的相似度对文档进行检索。实验结果表明，本文提出的方法是有效的，能够进一步提高检索的精度。本文对文本检索的语义信息进行了有益的尝试，实验的结果虽然对比传统方法有一定的改进，但以主谓宾结构进行衡量的方法计算句子之间的相似度还需要进一步优化和改进。今后我们将继续考察文本之间基于句子结构相似度的相关应用及其探索。

参 考 文 献

[1] Salton G, Wong A, Yang C S. A vector space model for automatic indexing[J]. Communications of the ACM, 1975, 18(11): 613-620

[2] Deerwester S, Landauer D S, et al. Indexing by latent semantic analysis[J]. Journal of the American Society for Information Science, 1990, 41: 391-407

[3] 郑廷，郑诚. 基于 Lucene 的语义检索系统[J]. 计算机工程，2008, 34(16): 92-94

[4] Fellbaum C, et al. WordNet: An Electronic Lexical Database [M]. MIT Press, 1998

[5] Du Lan, Jin Huidong, De Vel, et al. A latent semantic indexing and WordNet based information retrieval model for digital forensies[C]// IEEE International Conference on Intelligence and Security Informatics. IEEE ISI, 2008: 70-75

[6] Ravishankar D, Thirunarayan K, Immaneni T. A modular approach to document indexing and semantic search[C]// Proceedings of the IASTED International Conference on Web Technologies, Applications, and Services(WTAS 2005). 2005: 165-170

[7] Selvi P, Gopalan N P. Sentence Similarity Computation Based on WordNet and Corpus Statistics[C]// the Proceeding of International Conference on Computational Intelligence and Multimedia Applications. 2007: 9-14

[8] Aliguliyev R M. A new sentence similarity measure and sentence based extractive technique for automatic text summarization[J]. Expert Systems with Applications, 2009, 36: 7764-7772

[9] Shehata S, Karray F, Kamel M. A Concept-based Model for Enhancing Text Categorization[C]// Proceedings of KDD'2007. San Jose, California, USA, 2007: 629-637

[10] Pedersen T, Patwardhan S, Michelizzi J. Wordnet::similarity-measuring the relatedness of concepts[A]// Proc. of AAAI-04 [C]. San Jose, California, USA, 2004: 1024-1025

[11] Budanitsky A, Hirst G. Semantic distance in WordNet: an experimental, application-oriented evaluation of five measures[C]// 2nd Meeting of the North American Chapter of the Assoc for Computational Linguistics. Pittsburgh, PA, 2001

[12] Lin D. An information-theoretic definition of similarity [C]// Proceedings of the 15th International Conference on Machine Learning. Madison, WI, US, 1998

[13] LingPipe, Alias-i, Inc. http://www.alias-i.com

(上接第 136 页)

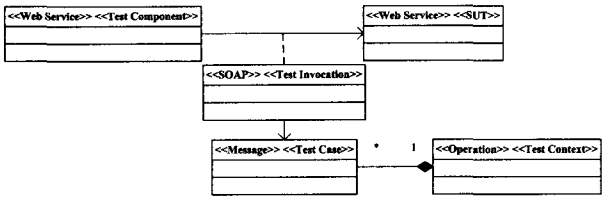

图 11 与 Test Architecture 和 Test Behavior 相关的 U2WSTP

U2TP 中的 SUT 的概念是指通过公共的、可用的接口提供的一组操作，而这与 Web Services 通过服务访问点对外提供服务的思想是类似的。由于在 U2TP 中，TestCase 本身也是从 Operation 和 Behavior 两个元类扩展而来的，因此上述衍型叠加表达了 Web Services 测试中的 Test Case 也是面向消息传输行为的，同时衍型元类之间的聚集关系也说明了 Test Context 作为一个 Operation 能对外提供接口。Test Case 是一种行为特征，它描述了 Web Services 之间(即 Test Components 和 SUT 之间)如何通过 SOAP 消息进行交互，以实现特定的测试目标。

与 Test Data 包有关的 U2WSTP，如图 12 所示。

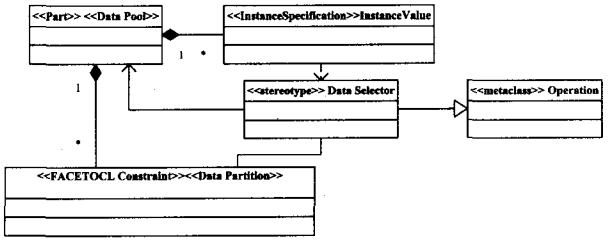

图 12 与 Test Data 相关的 U2WSTP

结束语 本文在引入两种主要的模型扩展方法的基础上，采用 Profile 对 UML 元模型进行扩展，形成针对 Web Services 的术语描述集，讨论各个包的扩展结果，运用 Stereotype的叠加技术实现了面向 Web Service 的 U2TP 测试模型扩展，描述了与 Test Architecture, Test Behavior 以及 Test Data 包相关的测试模型。本文的研究结果为模型驱动的 Web Service 的自动化测试奠定了基础。

参 考 文 献

[1] IBM Haifa Research Laboratory. Model Driven Testing Tools [R]. AGEDIS 1999-20218. Release 4.0.0, 2003

[2] The Object Management Group. Unified Modeling Language2.0 Infrastructure Specification[EB/OL]. http://www.omg.org/technology/documents/formal/uml.htm, 2005-07-05

[3] 蒋严冰，邵维忠，张路，等. UML 中衍型的精确定义与分析[J]. 电子学报，31(12A): 2101-2105

[4] UML 2.0 Testing Profile Specification[R]. OMG Adopted Specification(Version1.0) formal/05-07-07

[5] 王立军，白晓颖，刘如娟. 面向服务的软件可靠性探讨[J]. 小型微型计算机系统，2009(6): 1031-1037

[6] The Object Management Group, Unified Modeling Language2.0 Superstructure Specification[EB/OL]. http://www.omg.org/technology/documents/formal/uml.htm, 2005-07-04

[7] 柴晓路，梁宇奇. Web Services 技术、架构和应用[M]. 北京：电子工业出版社，2003

[8] Heckel R, Lohmann M. Towards Contract-based Testing of Web Services[J]. Theoretical Computer Science, 2004, 82(6)

[9] Triebsees T. Constraint-based Model Transformation: Tracing the Preservation of Semantic Properties[J]. Journal of Software, 2007, 2(3)