УДК 681.3.06:004.89

Ле Хоай, А.Ф. Тузовский

Поиск в семантических электронных библиотеках

Рассматриваются четыре вида поиска на основе использования семантических технологий в электронных библиотеках, анализируются возможные варианты их применения, объясняются решаемые с их помощью задачи и предлагаются новые методы реализации.
Ключевые слова: семантические технологии, семантическая электронная библиотека, семантический поиск, контекстные метаданные, контентные метаданные, поиск по графу.

Поиск информации представляет собой важную функциональность, которую предоставляют почти все информационные системы, и в особенности электронные библиотеки (ЭБ), автоматизирующие работу пользователей с электронными ресурсами, такими как документы, изображения, аудио- и видеофайлы.

В электронных библиотеках поиск информации считается одной из основных задач, решение которой позволяет пользователям находить требуемые им информационные ресурсы. В них предоставляется возможность выполнять полнотекстовый поиск (поиск терминов) для текстовых электронных документов или поиск в метаданных для других типов ресурсов.

Проблемой полнотекстового поиска является отсутствие понимания смысла информационных потребностей пользователей, смысла искомых текстовых документов. Выполнение полнотекстового поиска затрудняется в связи с такими проблемами, как синонимия, полисемия и омонимия. Это повлечет за собой снижение релевантности получаемых результатов поиска. Для их решения требуется обрабатывать документы и формализовать содержащуюся в них семантику с использованием различных моделей представления знаний (таких, как, например, таксономии, тезаурусы и онтологии). В настоящее время уже разработано много методов решения данной задачи, например на основе определения встречаемости простых наборов ключевых слов [1, 2] либо встречаемости ключевых слов из словарей предметных областей [3–5].

Для выполнения эффективной формализации семантики информационных ресурсов организация W3C предложила набор специальных семантических технологий (набор языков и стандартов). Например, RDF (Resource Description Framework) – для описания семантических данных (метаданных); RDFS, OWL (Ontology Web Language) – для описания семантических моделей; SPARQL (Simple Protocol And RDF Query Langue) – для описания запросов к таким данным [6]. Целью создания этих стандартов является предоставление средств для явного и формального описания семантики ресурсов в новой концепции Semantic Web.

Семантические электронные библиотеки (СЭБ), т.е. ЭБ, основанные на использовании семантических технологий, позволяют реализовать работу с электронными документами на новом уровне, в особенности при выполнении поиска, учитывающего семантику [7].

В данной статье рассматриваются четыре различных вида поиска ресурсов, такие как простой поиск, поиск по графу, семантический контекстный поиск и семантический контентный поиск, а также описан способ их применения в разрабатываемой авторами семантической электронной библиотеке. Для пояснения предложенных методов вначале необходимо пояснить используемый онтологический подход к описанию ресурсов и способы оценки их семантической близости.

Онтологический подход к созданию метаописания документов. Идея данного подхода заключается в использовании онтологий предметных областей для аннотирования содержания электронных документов [7, 8]. В СЭБ аннотирование электронных документов, содержащих знания из разных предметных областей, выполняется с использованием специально разработанных онтологий этих предметных областей, описанных с использованием таких языков, как RDFS или OWL. Под аннотированием ресурса понимается его семантическое метаописание в виде набора простых высказываний (триплетов), в состав которых могут входить контекстные и контентные семантические метаданные. Под простым высказыванием понимается структура вида $\langle s, p, o \rangle$, где s – субъект высказывания, p – предикат высказывания, а o – это объект высказывания.

Под контекстными метаданными аннотации объекта (документа, пользователя, проекты и т.п.) понимаются утверждения о его связи с другими объектами, понятиями из общих онтологий библиотеки, таких как онтологии пользователей или онтологии ресурсов.

Под контентными метаданными понимаются утверждения о знаниях, содержащихся в самом аннотируемом ресурсе. Наборы триплетов, составляющие контентные метаданные, создаются на основе онтологий, описывающих те предметные области, с которыми связан данный ресурс.

Определение 1. *Под онтологической моделью* (*онтологией*) *O понимается знаковая система* ⟨*C, P, I, L, T*⟩, *где C – множество элементов, которые называются понятиями; P – множество элементов, называемых свойствами* (*двуместными предикатами*)*; I – множество экземпляров понятий; L – множество текстовых меток или значений понятий и свойств; T – частичный порядок на множестве C и P.*

Любому свойству $p \in P$ может быть задано значение семантического веса $pv \in [0, 1]$, которое задает смысловую близость между субъектом и объектом утверждения, составленного с использованием данного свойства. При нулевом значении *pv*-субъект и объект простого высказывания считаются не связанными между собой по семантике, а если значение *pv* равно единице, то считается, что субъект и объект имеют одинаковую семантику.

На основе онтологической модели можно определить контекстные и контентные метаданные ресурсов.

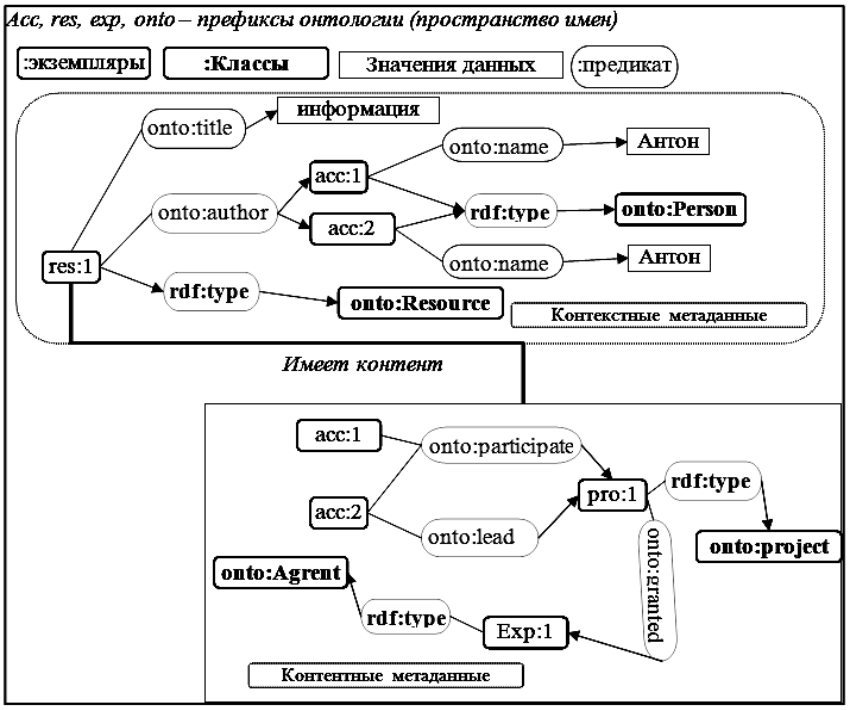

Рис. 1. Пример метаданных

Определение 2. *Контекстные метаданные ресурса s – это набор простых утверждений* (*триплетов*) *Мк* = {*t*1, *t*2, ..., *tn*}, *описывающих его связи с другими объектами или которые связаны с объектом, но не содержатся в его описании. Каждый триплет tj имеет вид tj* = (*s*, *p*, *o*), *где s – это аннотируемый ресурс* (*субъект*), *которому задан идентификатор* (*URI*) ($s \in I$), *o – объект утверждения* ($o \in C \cup I \cup L$), *p – отношение между субъектом и объектом* ($p \in P$).

Определение 3. *Контентные метаданные ресурса – это набор простых утверждений (кортежей) Мс* = {*t*1, *t*2, …, *tm*}, *описывающих основные знания, содержащиеся в документе. Каждый кортеж ti имеет вид ti* = (*s*, *p*, *o*, *v*), *где s – это субъект утверждения* ($s \in C \cup I$), *o – объект утверждения* ($o \in C \cup I$), *p – отношение между субъектом и объектом* ($p \in$ P), *а v – весовой коэффициент, который оценивает значимость данного утверждения*.

Простые утверждения, созданные на основе онтологии *O*, могут иметь следующий вид: <*C*, *P*, *C*>, <*I*, *P*, *I*>, <*I*, *P*, *C*>, <*I*, *P*, *L*>, при этом *C*, *P*, *I* представляются в виде уникальных URI-идентификаторов понятий, предикатов и экземпляров.

Семантические метаданные также могут быть графически представлены в виде направленного ациклического графа (directed acyclic graph, DAG) [2], как показано на рис. 1, где вершинами графа являются субъекты и объекты триплетов, а ребрами – их предикаты (отношения).

Семантическая близость. Под семантической близостью между элементами понимается их смысловое сходство. Семантическая близость может определяться между разными компонентами триплетов. При этом в качестве базовой близости можно рассматривать близость между элементами онтологий (классами, предикатами, терминами). Классы и предикаты являются элементами схемы онтологий, а термины – элементами онтологии предметных знаний в виде таксономии или тезауруса, представленных с использованием стандарта *SKOS* [13] (*Simple Knowledge Organization System*).

Например, семантическая близость может быть оценена для классов, показанных на рис. 2, таких как ***человек***, ***сотрудник***, ***студент*** и ***аспирант***, или для терминов – ***animal***, ***fish***, ***mammal*** и ***puma***.

На основе оценки семантической близости между элементами онтологий может быть определена и близость между метаданными ресурсов ЭБ.

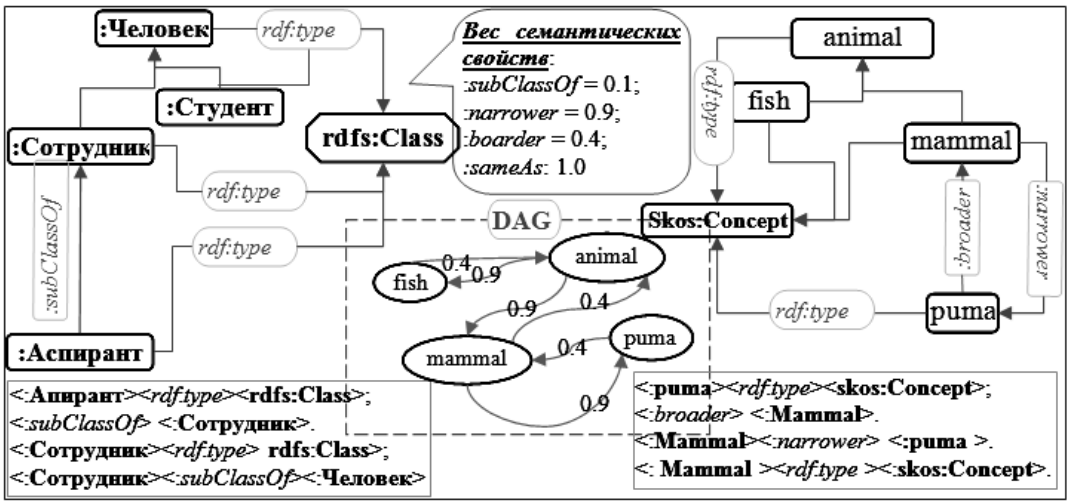

Рис. 2. Примеры оценки семантической близости

Следует отметить, что с учетом направления семантических отношений (от субъекта к объекту) *семантическая близость не является симметричной*. Допустим, необходимо вычислить близость между терминами *<mammal> и <puma> с использованием* отношения (*<mammal> <:narrower> <puma>*), как показано на рис. 2 и обозначено Sim(*mammal*, *puma*). Если под Sim(*mammal*, *puma*) понимается результат запроса на поиск всех *<mammal>*, то ответом могут быть *<mammal>* и *<puma>*, а если искать Sim(*puma*, *mammal*), то первым ответом не может быть *mammal.* Это означает, что Sim(*mammal*, *puma*) не равно Sim(*puma*, *mammal*). На рис. 2 показан пример того, что Sim(*mammal*, *puma*) равно значению семантического веса предиката *«:narrower»* (0,9), а Sim(*mammal*, *puma*) – *«:broader»* (0,4).

Предикаты (свойства) онтологий будем называть семантическими, если им будут заданы весовые коэффициенты *pv*, ∈ (0,1]. Например: предикаты *<:broader>*, *<:subClassOf>* являются семантическими (т.к. им заданы веса 0,9 и 0,4 соответственно), а предикат *<:rdf:type>* не является семантическим, т.к. ему вес не задается. Задание веса предикатов возлагается на создателей онтологий в соответствии с их пониманием онтологии и потребностями решаемых задач эмпирическим способом [3; 9. С. 85–87; 11]. При этом не следует путать весовые коэффициенты предикатов с весовыми коэффициентами триплетов в *определении 3*.

Для вычисления семантических близостей необходимо использовать GO-граф. Он строится в соответствии со следующими правилами:

- Используются только триплеты с семантическими предикатами (*pv* # 0).
- Вершинами графа являются субъекты и объекты триплетов, а ребра графа от субъекта к объекту триплета имеют вес, равный значению семантического веса (*pv*) предиката данного триплета.
- Симметричное семантическое отношение добавляет в граф два ребра с равными весами, например: *<owl:sameAs>* добавляет два ребра со значениями *pv* = 1,0.

Под путем PATH(a, b) между двумя вершинами a и b графа GO понимается набором ребер (предикатов), ведущих от вершины a до вершины b с учетом их направленности.

Пусть a и b – это элементы некоторой онтологии O. Тогда значение семантической близости Sim(a, b) между этими вершинами вычисляется следующим образом:

$$\mathrm{Sim}(a,b) = \max_{i=1\to k}\left(\mathrm{Sim}_{PATH_i}(a,b)\right), \quad (1)$$

где k – число возможных путей графа GO от вершины a до вершины b. Значение семантической близости между a и b по направлению пути i sim_{PATH_i} (a, b) определяется по следующей формуле:

$$\mathrm{Sim}_{PATH_i}(a,b) = \prod_{j=1}^{h_i} pv_{i,j}, \quad (2)$$

где h_i – число возможных семантических отношений между a и b на пути i; $pv_{i,j}$ – значение веса ребра на основе j-го семантического предиката на пути i. Из (1) и (2) можно получить окончательную формулу для определения семантической близости между вершинами a и b:

$$\mathrm{Sim}(a,b) = \max_{i=1\to k}\left(\mathrm{Sim}_{PATH_i}(a,b)\right) = \max_{i=1\to k}\left(\prod_{j=1}^{h_i} pv_{i,j}\right). \quad (3)$$

Sim(a, b) удовлетворяет следующим свойствам: Sim(a, b) $\in$ [0,1]; равно 0 при отсутствии пути от a к b; Sim(a, a) = 1.

В качестве примера можно рассмотреть вычисление семантической близости между терминами <*puma*> <*animal*> с использованием графа *DAG*, показанного на рис. 2. Между ними имеются две дуги (<*puma*> <*:broader* – 0,4> <*mammal*>) и (<*mammal*> <*:broader* – 0,4> <*animal*>). С использованием этих дуг можно построить два пути Sim(*puma*, *animal*) = 0,4×0,4 = 0,16 и Sim(*animal*, *puma*) = = 0,9×0,9 = 0,81. Таким образом, окончательная оценка семантической близости равна максимальной из них – 0,81.

На основе оценки семантической близости между базовыми элементами можно уже определять семантическую близость между триплетами, экземплярами и между наборами триплетов. Такие оценки будут рассматриваться, использоваться при описании разных видов семантических поисков. Другие методы определения различных видов семантической близости описаны в [9].

Виды поиска в семантической ЭБ. В СЭБ все информационные объекты (ИО – электронные ресурсы, пользователи, категории и др.) описываются метаданными, представленными с помощью *RDF-триплетов* на основе использования элементов некоторых онтологий. Онтологии с их экземплярами формируют базу знаний (*KB – Knowledge Base*) СЭБ. В *KB* могут выполняться запросы, описанные с помощью языка *SPARQL* [12].

Простой поиск. Поиск информационных объектов на основе использования их лексических меток.

Пусть каждый ИО s описывается в базе знаний KB метаданными M_L, состоящими из m триплетов, которые определяют его интерпретацию на естественных языках, и обозначается на основе онтологий O:

$$M_L(s) = (p_1(s, l_1) \vee p_2(s, l_2) \vee \ldots \vee p_m(s, l_m)),$$

где каждое утверждение $p_j(s, l_j)$ (соответствующее RDF-триплету (s, p_j, l_j)) состоит из субъекта $s \in C \cup I$ отношения $p_j \in R$, для которого определяются метаданные и значения $l_j \in L$.

Постановка задачи: Требуется найти все ИО в KB с заданными текстовыми метками $l \in L$.

Решение: Результатом данного запроса должен быть любой ИО с его набором M_L, в котором содержится хотя бы одно значение l_j из лексических меток, удовлетворяющее следующее условие*:*

$$\text{Подстрока}(l_j) = l \quad (4)$$

Решение данной задачи заключается в формировании в KB *SPARQL*-запроса, который содержит только один известный параметр l.

Допустим, в KB на рис. 1 для интерпретации документа используется предикат <*:title*>, а автора – <*:name*>. Кроме этих предикатов, могут использоваться и другие.

Решение: Данная задача может быть решена следующим образом (пояснение выполнено на основе рис. 1).

1. Создание нового предиката, для которого все свойства (<*:title*>, <*:name*> *и др.*), *выполняющие текстовое описание, являются подсвойствами нового созданного свойства* (например: <*:hasName*>) *с помощью отношения* (*rdfs:subProperty*)*;*

/* здесь комментарий*/

/* *<:hasName>* – новое созданное отношение*/

/*Добавленные триплеты: */

⟨:title, rdfs:subProperty, :hasName⟩

⟨:name, rdfs:subProperty, :hasName⟩

..... /* другие добавленные триплеты при необходимости*/

2. *Составление SPARQL-запроса на основе триплета с созданым свойством для поиска в KB с логическим выводом;*

/* <?X> - искомый параметр*/

/*Запрос на языке SPARQL: */

*SELECT * FROM*

?*IO* :*hasName* ?*L*, /*?IO – сам искомый объект, ?L – его название*/

?*IO rdf:type* ?T /* ?T – его тип*/

... /* включить другие отношения */

WHERE ?*L LIKE* "**l*" /* условие запроса для фильтрации*/

3. *Получение набора ИО с их названиями и типами или др.*

Семантика каждого информационного объекта в *KB* описывается явно, т.е. он имеет тип и отношения с другими объектами. В связи с этим в задаваемый SPARQL-запрос можно включать разные значимые отношения (например, тип объекта и т.п.), которые помогают пользователю уточнить требуемый ответ.

Поиск по графу. Данный вид поиска похож на простой поиск, который использует запрос *SPARQL* для решения задачи. Все отличие будет заключаться в том, что в запросе ***идентификатор*** (*URI*) рассматриваемого объекта будет ***известен***.

Пусть для рассматриваемого объекта *s* в *KB* имеется конечный набор из *h* триплетов его контекстных метаданных и обозначается как: $M_k(s)=\{t_i=\langle s, p_i, o_i\rangle \vee\ i=1\div h\}$, где $s\in I$, $p_j\in R$ и $o_j\in C\cup I\cup L$.

Постановка задачи: Для некоторого объекта *s*, имеющего набор отношений $p_j(s, o_j)$, требуется найти все объекты o_j, которые связаны с *s* отношением p_k и обладают заданными свойствами p_n. Поиск по графу может повторяться для задания $s = o_j$.

Например: нужно **найти всех друзей Бориса, живущих в городе Томске и обучающихся в ТПУ**. Здесь *Борис* – это объект *s*, *Друзья* – это объекты *о на основании* отношения p_k, а ⟨*Живут* и *Обучаются*⟩ – свойства (отношения) p_n.

Решение задачи: Для выбранного пользователем отношения p_k и для заданных отношений (свойств) p_n формируется соответствующий им запрос в KB на основе языка SPARQL. При выборе пользователя отправляется его соответствующий запрос в KB и полученные результаты показываются пользователям.

Например, на рис. 1 для объекта проекта *<:pro1>* система может предложить для поиска отношение **(<***onto:paticipate***> –** *участвующие***)** и свойства *<onto:profession>* для выполнения такого запроса, как найти всех участников **(<***onto:paticipate***>) проекта** *<:pro1>*, имеющих профессию *Программист*. Имеется следующий шаблон запроса на языке *SPARQL*:

```
/*Запрос на языке SPARQL: */
SELECT * FROM
S :pk ?O,      /*?O – сам искомый объект, :pk – выбранное
отношение для S*/
?O :p1 ?Oi,       /* ?T – его тип*/
..., /* включить другие отношения */
?O :pn ?Oi+1
WHERE  /* условие запроса для фильтрация*/
```

Простой поиск и поиск по графу не учитывают семантические близости, в связи с этим они не обеспечивают ранжирование результатов. Ранжирование может быть осуществлено при контекстном и контентном поиске.

Контекстный поиск. Контекстный поиск объектов выполняется путем выполнения поискового запроса к контекстным метаданным. Примером такого поиска может служить поиск проектов, документов и других информационных объектов, которые включают заданный набор отношений.

Постановка задачи: требуется найти объекты (проект, документ, и др.) с заданным шаблонным запросом $Q_{context} = \{t_i = \langle s_q, p_i, o_i\rangle \vee i \in [1, k]\}$ и проранжировать полученные результаты по убыванию семантической близости.

Решение: Пусть в КВ имеется конечный набор $\boldsymbol{M_{PR}}$ возможных объектов, которые удовлетворяют заданному запросу $Q_{context}$, тогда результатом будет любой объект $s_r \in M_R \subseteq M_{PR}$ с набором триплетов его контекста $R_{context} = \{t_j = \langle s_r, p_j, o_j\rangle \vee j \in [1, h]\}$, если значение семантической близости $\mathrm{Sim}_{context}$ между $Q_{context}$ *и* $R_{context}$ будет удовлетворять следующему условию:

$$\mathrm{Sim}_{context}\left(Q_{context}, R_{context}\right) > \varepsilon, \tag{5}$$

где ε – заданное пороговое значение. В контекстном поиске s_q и s_r не участвуют в формуле (5), так как s_q – искомая неизвестная переменная, а используются только компоненты p и o их триплетов $Q_{context}$ и $R_{context}$. Для вычисления выражения (5) на основе семантической близости между экземплярами может быть использована следующая формула:

$$\mathrm{Sim}_{context}\left(Q_{context}, R_{context}\right) = \frac{\sum_{i=1}^{k} \max_{j=1\to h}\left(\mathrm{Sim}(t_i, t_j)\right)}{k} > \varepsilon, \tag{6}$$

где $\mathrm{Sim}(t_i, t_j)$ определяется следующим образом:

$$\mathrm{Sim}(t_i, t_j) = \mathrm{Sim}(p_i, p_j) \times \mathrm{Sim}(o_i, o_j), \tag{7}$$

где $\mathrm{Sim}(p_i, p_j)\, u\, \mathrm{Sim}(o_i, o_j)$ – семантические близости между предикатами и объектами триплета t_i и t_j, соответственно, которые вычисляются по формуле (3), $\mathrm{Sim}_{context}$ – значение семантической близости между экземплярами (контекстами).

Результаты поиска упорядочиваются по убыванию значений семантических близостей, вычисленных по формуле (6).

Контентный поиск. Контентный поиск применяется для объектов, проаннотированных наборами триплетов контентных метаданных.

Постановка задачи: Найти информационные объекты, соответствующие шаблонному запросу $Q_{content} = \{t_i = \langle s_i, p_i, o_i\rangle \vee i \in [1, k]\}$, и проранжировать полученные результаты по убыванию семантической близости.

Решение: Пусть в КВ имеется конечный набор $\boldsymbol{N_{PR}}$ возможных объектов, которые удовлетворяют заданному запросу $Q_{content}$, и тогда результатом будет любой объект $S_R \in N_R \subseteq N_{PR}$ с набором триплетов его контента $R_{content} = \{t_j = \langle s_j, p_j, o_j\rangle \vee j \in [1, h]\}$ при условии того, что значение семантической близости $\mathrm{Sim}_{content}$ *между* $Q_{content}$ *и* $R_{content}$ *удовлетворяет следующему условию*:

$$\mathrm{Sim}_{content}\left(Q_{content}, R_{content}\right) > \varepsilon, \tag{8}$$

где ε – заданное пороговое значение. Разложение левой части формулы (8) похоже на формулу (6) и отличается только тем, что в контентном поиске в вычислении семантической близости участвуют все компоненты триплетов из контентных метаданных. Для вычисления семантической близости между триплетами t_i, t_j может быть использована следующая формула:

$$\mathrm{Sim}(t_1, t_2) = \begin{cases} |k| \dfrac{\mathrm{Sim}(s_1, s_2) + \mathrm{Sim}(o_1, o_2)}{2}, \vee k > 0, \\ |k| \dfrac{\mathrm{Sim}(s_1, o_2) + \mathrm{Sim}(o_1, s_2)}{2}, \vee k \le 0, \end{cases} \tag{9}$$

где $k = \mathrm{Sim}(p_1, p_2)$ и $\mathrm{Sim}(t_1, t_2) \in [0, 1]$, $\mathrm{Sim}(t_1, t_2) = 0$ при $k = 0$. При $k < 0$ два предиката p_1 и p_2 имеют отношение $\langle p_1, owl{:}inverseOf, p_2\rangle$, при наличии триплета $\langle s, p_1, o\rangle$ подразумевается триплет $\langle o, p_2, s\rangle$. Семантическая близость между компонентами триплета t_i и t_j вычисляется по формуле (3), а весовые коэффициенты всех утверждений контентных метаданных одинаковы ($v = 1$).

Результаты поиска ранжируются по убыванию семантических близостей, вычисленных по формуле (8).

На практике наборы M_{PR} и N_{PR} формируются путем фильтрации объектов данного типа с помощью языка SPARQL [10] с целью ограничения количества объектов, для которых вычисляется семантическая близость.

Реализация видов поиска в СЭБ. Все рассмотренные виды поиска являются полезными в семантических ЭБ для повышения точности поиска и использования возможностей семантических технологий.

Примеры реализации интерфейса простого вида поиска, позволяющего пользователям составлять набор триплетов для поиска по контенту и контексту, показан на рис. 3 и 4.

Все показанные примеры интерфейсов реализованы в разрабатываемой авторами семантической электронной библиотеке *SemDL*. Данная библиотека предоставляет возможность составлять триплеты запроса для контентных и контекстных метаданных базы знаний системы.

Для контекстного поиска вначале необходимо выбрать тип объекта, а затем можно составлять другие триплеты о контексте данного объекта. Возможные свойства (предикаты) будут определяться автоматически для выбранного типа объекта и предлагаться пользователю.

Результаты поиска ранжируются по убыванию значения семантической близости между набором триплетов искомого объекта и набором триплетов релевантных объектов.

Проводилась оценка точности работы предлагаемых видов контентного и контекстного поиска с использованием экспертов [3, 9]. В результате были получены достаточно высокие оценки точности и полноты, примерно 85%. В некоторых случаях точность достигает 100% при учете *строгой несимметричности* семантических близостей между базовыми элементами.

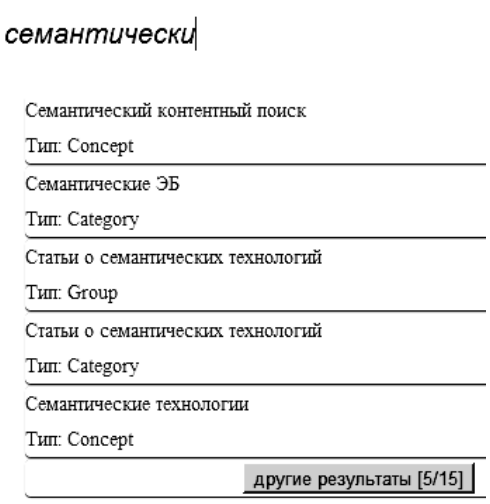

Рис. 3. Пример интерфейса простого поиска

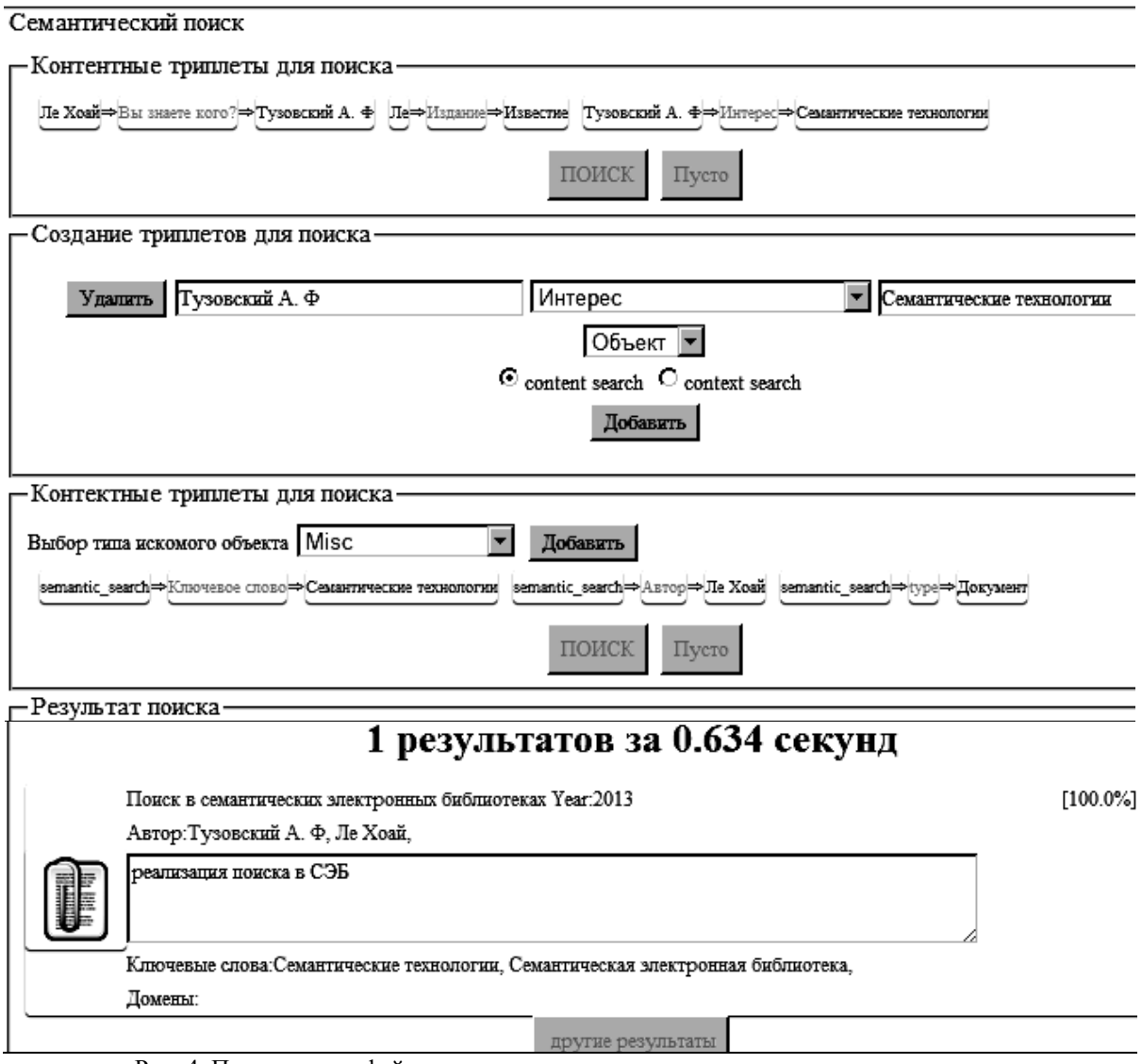

Рис. 4. Пример интерфейса для выполнения контекстного и контентного поиска и результата поиска по контексту

Заключение. Поиски по контенту и контексту позволяют ранжировать результаты поиска на основе их значений семантической близости. Контентный поиск применяется для поиска по содержанию объектов, контекстный поиск используется для поиска всех объектов системы на основе их контекстных метаданных. В настоящее время широкое применение получили простой поиск и поиск по графу, по сравнению с остальными видами, так как они не требуют вычислений семантических близостей. Однако использование контекстного и контентного поиска позволяет выполнять более сложные запросы и получать более точные результаты.

Литература

1. Нгуен Б.Н. Классификация текстов на основе оценки семантической близости терминов / Б.Н. Нгуен, А.Ф. Тузовский // Изв. Том. политехн. ун-та. – 2012. – Т. 320, № 5. – С. 43–48.

2. Taylor William P. A comparative study on ontology generation and text clustering using VSM, LSI, and document ontology models. – Clemson University, 2007. – 58 p.

3. Saša Neši´. Semantic Document Architecture for Desktop Data Integration and Management: Doctoral Dissertation. – Italia, 2010. – 228 p.

4. Hotho A. Wordnet improves Text Document Clustering /A. Hotho, S. Staab, and G. Stumme // Proceedings of the SIGIR 2003 Semantic Web Workshop. – 2003. – P. 541–544.

5. Using WordNet Similarity and Antonymy Relations to Aid Document Retrieval [Электронный ресурс]. – Режим доступа: http://www-users.cs.york.ac.uk/kazakov/papers/desimone-kazakov-crc.pdf (дата обращения: 01.03.2013).

6. Hendler A. J. Handbook of Semantic Web Technologies. – Springer, 2011. – 1077 p.

7. Ле Х.Х. Разработка электронных библиотек на основе семантических технологий // Научно-технический вестник Поволжья (Казань). – 2012. – № 3. – С. 138–145.

8. Тузовский А.Ф. Формирование семантических метаданных для объектов системы управления знаниями // Изв. Том. политехн. ун-та. – 2007. – Т. 310, № 3. – С. 108–112.

9. Нгуен Б.Н. Модели и методы поиска информационных ресурсов с использованием семантических технологий: дис. … канд. техн. наук. – Томск, 2012. – 198 с.

10. Ле Х. Формирование рекомендаций в семантических электронных библиотеках / Хоай Ле, А.Ф. Тузовский // Проблемы информатики. – 2012. – № 3. – С. 113–119.

11. Gong Z. Multi-term Web Query Expansion Using WordNet / Z. Gong, C.W. Cheang, U.H. Leong // DEXA'06 Proceedings of the 17th Database and Expert Systems Applications Conference. – Springer-Verlag Berlin, Heidelberg, 2006. – P. 379–388.

12. SPARQL 1.1 overview [Электронный ресурс]. – Режим доступа: http://www.w3.org/TR/2012/WD-sparql11-overview-20120501/, свободный (дата обращения: 01.03.2013).

13. SKOS Simple Knowledge Organization System RDF Schema [Электронный ресурс]. – Режим доступа: http://www.w3.org/TR/2008/WD-skos-reference-20080829/skos.html (дата обращения: 01.03.2013).

Ле Хоай
Аспирант каф. оптимизации систем управления НИ ТПУ
Тел.: +7-913-108-01-44
Эл. почта: lehotomsk@yahoo.com

Тузовский Анатолий Федорович
Д-р техн. наук, профессор каф. оптимизации систем управления НИ ТПУ
Тел.: +7 (382-2) 42-14-85
Эл. почта: tuzovskyaf@tpu.ru

Hoai Le, Tuzovsky A.F.
Search in semantic digital libraries

In this paper four types of search in digital libraries on the basis of using semantic technologies are considered, possible options of their application are analyzed, tasks solved with their help are explained and new methods of implementation are proposed.
Keywords: semantic technology, semantic digital library, semantic search, contextual metadata, content metadata, graph search.