OSTIS-2013
(Open Semantic Technologies for Intelligent Systems)

УДК 004.822:514

ОНТОЛОГИЧЕСКИ-ОРИЕНТИРОВАННАЯ СИСТЕМА КЛАСТЕРИЗАЦИИ И ПОЛНОТЕКСТОВОГО ПОИСКА ПРОЕКТНЫХ ДОКУМЕНТОВ

Наместников А.М.[*], Субхангулов Р.А.[*], Филиппов А.А.[*]

[*] *Ульяновский государственный технический университет, г. Ульяновск, Россия*

nam@ulstu.ru

subkhangulov-ruslan@yandex.ru

al.filippov@ulstu.ru

В статье представлено описание программной системы, которая позволяет выполнять кластеризацию проектных текстовых документов, основываясь на текущем состоянии предметной области в виде прикладной онтологии. Показана архитектура системы, структура онтологии и ее фрагменты в формате RDF, перечислены основные функции системы. Приведены результаты экспериментальных исследований, доказывающие эффективность разработанной программной системы.

Ключевые слова: интеллектуальная система; онтология; индексирование; кластеризация; полнотекстовый поиск.

Введение

В настоящее время во многих проектных организациях фактически завершен перевод архива проектной документации (ПД) в электронный формат. В связи с этим возникла необходимость в систематизации и автоматизации работы с электронным архивом ПД. Формирование к электронному архиву запросов допускается с использованием формализованного языка (такого, например, как SQL) при заранее известных атрибутах: децимальный номер, дата создания документа, автор и т.п. При таком подходе к построению архива ПД у проектировщика отсутствует возможность решать слабоформализованные задачи поиска. Такими задачами могут быть полнотекстовый поиск документа, нахождение близкого по содержанию документа, кластеризация всего множества документов и другие. Для решения подобного рода задач применяются интеллектуальные системы, функционирование которых основано на предметно-ориентированных знаниях. Эти знания могут быть представлены в виде онтологии предметной области [Добров и др., 2006].

В данной статье представлено описание программной системы, основными функциями которой является построение с использованием модели разметки Resource Description Framework (RDF) предметно-ориентированной онтологии, индексирование и кластеризация ПД на основе созданного онтологического описания предметной области и полнотекстовый информационный поиск.

1. Архитектура системы и модель онтологии

Система кластеризации и поиска ПД обладает следующими функциями:

- хранение и обработка документов в формате XML;
- создание и редактирование онтологии на основе моделей RDF и RDFS;
- онтологически-ориентированное индексирование ПД;
- онтологическая кластеризация документов;
- онтологический полнотекстовый поиск документов.

На рисунке 1 представлена архитектура разработанной системы. В качестве хранилища XML-документов используется XML-ориентированная СУБД Tamino, а в качестве хранилища онтологий Web-фреймворк Sesame.

Формально онтологию представим следующим образом:

$$O = < r, T, S, C, W, R >, \quad (1)$$

где r – корневая вершина онтологии, соответствующая классу проектных документов;

$T = \{t_1, t_2, ..., t_n\}$ – множество типов проектных документов ИПР, t_i – i-й тип проектного документа;

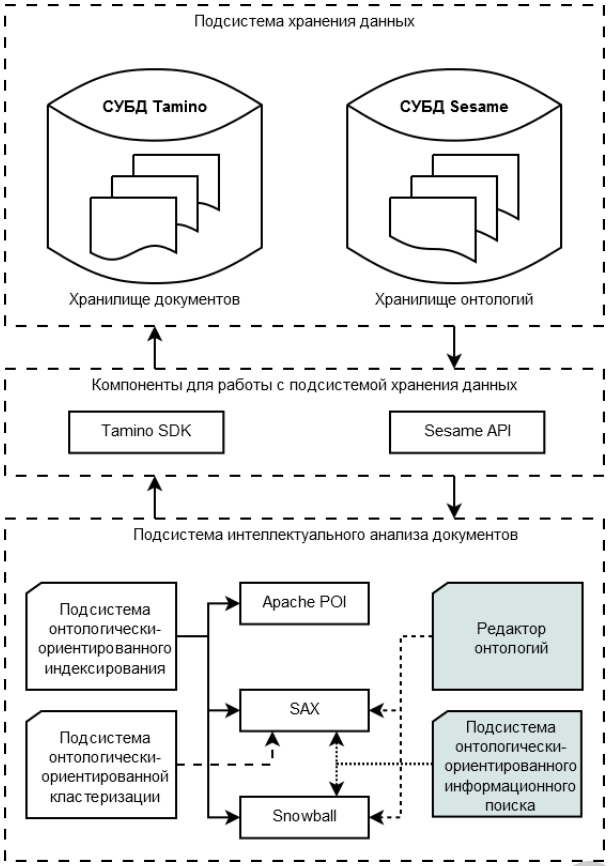

Рисунок 1 – Архитектура системы кластеризации и поиска ПД

$S = S^1 \bigcup S^2 \bigcup ... \bigcup S^n$ – множество структур документов;

$C = \{c_1, c_2, ..., c_k\}$ – множество понятий предметной области ИПР;

$W = \{w_1, w_2, ..., w_l\}$ – множество терминов предметной области ИПР.

R – множество отношений, определяемое следующим образом:

$$R = R_G \bigcup R_C \bigcup R_A,$$

где R_G – антисимметричное, транзитивное, нерефлексивное бинарное отношение обобщения;

R_C – бинарное транзитивное отношение композиции («часть–целое»);

R_A – конечное множество ассоциативных отношений.

Для структур документов справедливо соотношение:

$$\forall t_i \exists S^j : i = j.$$

Другими словами, для каждого типа документа в онтологии определена его структура.

Множество S^j содержит разделы и подразделы проектного документа типа t_j. В общем случае имеет место следующее неравенство:

$$S^i \bigcap S^j \neq \emptyset,$$

что означает допустимость пересечения структурных элементов между различными типами документов.

Онтология предметной области включает в себя два уровня: концептуальный и терминологический [Добров и др., 2006]. Концептуальный уровень представляется в виде дерева

$$(C, E),$$

где $C = \{c\}$ – множество концептов (понятий) предметной области, зафиксированных в онтологии;

$E = \{< c_i, c_k >\}, < c_i, c_k > \in C^2$ – множество дуг, соединяющих понятия.

Терминологический уровень для k-го понятия записывается в виде множества

$$\{(w_1^k, f_1^k), (w_2^k, f_2^k), ..., (w_i^k, f_i^k), ..., (w_{l_k}^k, f_{l_k}^k)\},$$

где w_i^k – i-й терм k-го понятия онтологии;

l_k – общее количество термов, ассоциированных с k-м понятием;

f_i^k – частота встречаемости i-го терма в описании k-го понятия.

2. Функциональные возможности системы

В редакторе онтологии реализованы следующие модули системы:

- Модуль формирования схемы онтологии и модуль формирования содержания онтологии предоставляют пользователю интерфейс для создания схемы онтологии и набора экземпляров классов онтологии.
- Модуль формирования терминологического окружения понятий онтологии реализована в модуле онтологической индексации.
- Модуль взаимодействия с RDF-хранилищем Sesame, где реализованы функции, позволяющие сохранять онтологию в хранилище Sesame и загружать онтологию из данного хранилища.
- Модуль визуализации онтологии позволяет производить редактирование создаваемой онтологии, представляя ее в виде графа, внешний вид которого можно настраивать (форму узлов, дуг и цветовую гамму).

Фрагмент созданной онтологии предметной области «Проектирование информационных систем» для интеллектуального анализа ПД имеет следующий вид:

<Conceptrdf:ID="Субъект"/>

<Conceptrdf:ID="Эксперт">
<Generalization rdf:resource="#Субъект"/>
</Concept>
<Concept rdf:ID="Проектировщик">
<Generalization rdf:resource="#Субъект"/>
</Concept>
<Concept rdf:ID="Тестировщик">
<Generalization rdf:resource="#Субъект"/>
</Concept>
<Concept rdf:ID="Программист">
<Generalization rdf:resource="#Субъект"/>
</Concept>
<Conceptrdf:ID="Объект"/>.

Модуль визуализации онтологии предназначен для получения изображения онтологии в виде ориентированного графа [Берштейн и др., 2005]. В качестве вершин графа изображаются понятия онтологии, а в качестве ребер – отношения между понятиями. Для любого понятия онтологии можно получить список всех ассоциированных с ним термов. Данная информация может быть представлена в виде таблицы, либо в виде дополнительного графа. Из полученной таблицы имеется возможность выбрать любой терм и система выделит на графе все понятия, с которыми он связан.

Индексация проектных документов состоит из следующих этапов:

- загрузка документов;
- анализ структуры документов;
- удаление стоп-слов;
- стемминг (выделение основы слова, получение термов);
- подсчет относительной частоты встречаемости термов;
- расчет степени выраженности понятий онтологии, построение онтологического представления для разделов и документов;
- генетическая оптимизация онтологических представлений.

В качестве входных данных подсистемы онтологически-ориентированной индексации выступают проектные документы в формате XML. Для данной подсистемы объектом обработки служит не целый ПД, а каждый его раздел в отдельности [Наместников 2009, Наместников и др., 2010]. Онтологическим представлением ПД считается такое описание ПД, которое состоит из множества понятий онтологии с соответствующими степенями выраженности данных понятий в документе.

В процессе онтологически-ориентированного индексирования ПД необходимо определить набор понятий предметной области, который содержится в тексте анализируемого документа.

Степень выраженности понятия c_k в j-м разделе ПД d будем вычислять по следующей формуле [Наместников и др., 2010]:

$$\mu_{S_j^d}(c_k) = 1 - \frac{1}{\sum_{s=1}^{l_k} \max\left(f_s^k, f_s^j\right)} \sum_{s=1}^{l_k} \left| f_s^k - f_s^j \right|,$$

где S_j^d – j-й раздел ПД d;

f_s^j, f_s^k – частоты встречаемости термина s в j-м разделе документа и в описании k-го понятия онтологии соответственно;

l_k – мощность текстового входа понятия c_k. В том случае, если термин s отсутствует в j-м разделе документа d, f_s^j принимается равным нулю.

В основе идеи оптимизации онтологического представления ПД лежит гипотеза – *любой текстовый документ можно разделить на множество непересекающихся фрагментов, в каждом из которых будет доминировать то или иное понятие предметной области*.

Пусть имеется предобработанный текстовый документ d, состоящий из последовательности термов:

$$S^d = w_{11}^d, w_{21}^d, ..., w_{i_1 1}^d, ..., w_{n_1 1}^d, w_{12}^d, ..., w_{i_2 2}^d, ..., w_{n_2 2}^d, ..., w_{i_j j}^d, ..., w_{n_m m}^d, \quad (2)$$

где i_j – номер терма в k-м предложении, $j = \overline{1,m}$;

$i_j = \overline{1,n_j}$, где n_j – количество термов в j-м предложении.

Обозначим через S_p^d часть последовательности S^d, которая определяется выражением (2), и запишем ее следующим образом:

$$S_p^d = w_{1p}^d, w_{2p}^d, ..., w_{j_p p}^d, ..., w_{n_p p}^d, \; p = \overline{1,s},$$

при этом выполняется равенство:

$$S_1^d, S_2^d, ..., S_s^d = S^d, \quad (3)$$

Для нахождения значения доминирования концептов будем применять метод сравнения текстового входа каждого понятия в онтологии предметной области с анализируемым текстом.

Алгоритм вычисления степени доминирования понятия в текстовом фрагменте состоит из следующих шагов:

Шаг 1. Определение максимальной степени выраженности концептов в текстовом фрагменте:

$$\hat{\mu}_{S_p^d}(c) = \max_c(\mu_{S_p^d}(c)).$$

Шаг 2. Определение среднего значения степени выраженности концептов онтологии, исключая концепт с максимальной степенью выраженности (определенный на предыдущем шаге):

$$\tilde{\mu}_{S_p^d}(c) = \frac{1}{n-1}\sum_{i=1}^{n-1}\mu_{S_p^d}(c_i),$$

где $c_i \in c - c_k$, $c_k = \arg\max_c(\mu_{S_p^d}(c))$, n – количество концептов с ненулевой степенью выраженности для текстового фрагмента S_p^d.

Шаг 3. Определение степени детерминированности понятия в текстовом фрагменте S_p^d:

$$\Delta_{S_p^d}(c) = \hat{\mu}_{S_p^d}(c) - \tilde{\mu}_{S_p^d}(c), \qquad (4)$$

Выражение (4) фактически определяет качество выделения текстового фрагмента в ПД с целью ограничения в тексте определенного понятия предметной области, которое зафиксировано в онтологии интеллектуального проектного репозитория.

Канонический генетический алгоритм характеризуется следующими особенностями [Скурихин и др., 1995]:

1. Задается целевая функция, определяющая эффективность найденного решения.
2. В соответствии с определенными ограничениями инициализируется исходная популяция потенциальных решений.
3. Каждая хромосома в популяции декодируется в необходимую форму для последующей оценки и затем ей присваивается значение эффективности в соответствии с целевой функцией.
4. Каждой хромосоме присваивается вероятность воспроизведения, которая зависит от эффективности данной хромосомы.
5. В соответствии с вероятностями воспроизведения создается новая популяция хромосом, причем с большей вероятностью воспроизводятся наиболее эффективные элементы.

Формально генетический алгоритм можно описать следующим образом [Скурихин и др., 1995]:

$$ГА = (P^0, \lambda, l, \nu, \rho, F, \tau),$$

где $P^0 = (a_1^0, ..., a_\lambda^0)$ – исходная популяция, где

a_i^0 – решение задачи, представленное в виде хромосомы;

λ – целое число (размер популяции);

l – целое число (длина каждой хромосомы популяции);

ν – оператор отбора;

ρ – отображение, определяющее рекомбинацию (кроссинговер, мутация);

F – целевая функция;

τ – критерий остановки.

Для решения конкретной задачи оптимизации текстовых фрагментов ПД генетический алгоритм требует следующих уточнений:

- способа кодирования хромосом (потенциальных решений);
- вида целевой функции;
- реализации операций кроссинговера и мутации.

Целью генетической оптимизации в процессе концептуального индексирования ПД является нахождение такой последовательности (3), которая соответствует минимальному значению целевой функции

$$F(S^d) = \frac{1}{s}\sum_p (1 - \Delta_{S_p^d}(c)) \to \min, \qquad (5)$$

$p = \overline{1,s}$, где s – количество текстовых фрагментов; $s = \overline{1,m}$, где m – количество предложений в индексируемом документе.

Таким образом, минимальный текстовый фрагмент соответствует одному отдельно взятому предложению ПД, а максимальный – целому ПД.

Потенциальное решение (хромосома) генетического алгоритма концептуального индексатора имеет следующий вид:

$$a_i^t = (< p, j >),\ p = \overline{1,s},\ j = \overline{1,m}, 1 \le s \le m, \qquad (6)$$

где p – номер текстового фрагмента;

j – номер предложения;

s – количество текстовых фрагментов;

m – количество предложений;

i – номер хромосомы;

t – номер поколения.

Таким образом, хромосома, определяемая выражением (6), представляет собой, в действительности, последовательность текстовых фрагментов (3).

Целевая функция определяет способ отображения хромосомы на единичный отрезок:

$$F : a_i^t \to [0,1].$$

В качестве целевой функции F будем использовать выражение (5).

На первом шаге работы генетического алгоритма формируется начальная популяция хромосом $P^0 = (a_1^0, ..., a_\lambda^0)$. Для каждой хромосомы a_i^0 определяется значение целевой функции $F(a_i^0)$. Затем производится ранжирование хромосом. Ранг элементов популяции $rank$ задается следующим образом:

$$\forall i \in \{1, ..., \lambda\} : rank(a_i^t) = i,$$

если для $\forall j \in \{1, ..., \lambda - 1\} : F(a_j^t) < F(a_{j+1}^t)$.

Первые g хромосом без изменения переходят в следующий пул (поколение), а остальное количество $(\lambda - g)$ формируется посредством операции кроссинговера. При определении оператора кроссинговера будем учитывать то, что последовательность предложений в тексте и их количество должны оставаться неизменными в процессе трансформации хромосом. Точка кроссинговера определяется случайным образом на границе двух текстовых фрагментов:

$$a_i^0 = (..., < p, j >, < p+1, j+1 >, ...)$$

для первой из двух хромосом, участвующих в кроссинговере. Так как в процессе рассматриваемой операции происходит взаимообмен частями хромосом и, принимая во внимание вышеприведенные ограничения, точку кроссинговера для второй хромосомы выбираем так, чтобы в левой части остались j первых предложений, как и у первой хромосомы.

Заключительным этапом формирования новой популяции является применение оператора мутации. В задаче концептуального индексирования предлагается применять два варианта мутации хромосом: 1) сдвиг границы текстового фрагмента и 2) объединение текстовых фрагментов.

Первый вариант мутации со сдвигом границы текстового фрагмента предполагает вероятностный выбор границы между двумя текстовыми фрагментами ПД. Далее принимается решение о направлении сдвига границы в правую или в левую сторону, учитывая количество предложений в соседних текстовых фрагментах. Граница перемещается на одно предложение в сторону с большим количеством предложений. При равенстве предложений направление выбирается случайным образом. Сдвиг границы не происходит в случае, если текстовый фрагмент содержит одно предложение.

Вариант мутации, объединяющий два соседних фрагмента, фактически уменьшает количество текстовых фрагментов в ПД за счет их укрупнения.

В основе подсистемы онтологически-ориентированной кластеризации ПД лежит модифицированный алгоритм нечеткой кластеризации Fuzzy C-Means, с учетом рассмотрения онтологического представления ПД как иерархии. Тем самым, мера сходства между ПД находится через сложность превращения одной иерархии в другую [Загоруйко, 1999].

В системе есть возможность учитывать две меры сходства ПД:

- мера сходства содержимого ПД;
- мера сходства структур ПД.

Подсистема онтологически-ориентированной кластеризации работает со следующими представлениями ПД: онтологическое представление документа как нечеткий вершинный подграф онтологии, оптимизированное онтологическое представление и классическое представление ПД в виде множества пар «термин−частота».

Подсистема онтологически-ориентированного информационного поиска состоит из следующих модулей:

- Модуль взаимодействия с онтологией − осуществляет процедуру подключения к онтологии, выполнение запросов к ней и обработку полученных результатов.
- Модуль поиска наиболее выраженного понятия − выполняет поиск в прикладной онтологии наиболее значимого понятия для набора ключевых слов (запроса пользователя).
- Модуль поиска терминов − для заданного понятия выполняет поиск дополнительных терминов в онтологии, которые в большей степени соответствуют найденному понятию.

Модуль поиска документов − осуществляет поиск среди множества документов по расширенному набору терминов.

3. Результаты экспериментов

Тестовое множество состоит из 262 проектных документов. Эксперт разбил данную выборку по четырем признакам:

- по классу документации (2 группы);
- по виду документации (14 групп);
- по разделу документации (14 групп);
- по тематике работ (38 групп).

Для оценки качества онтологически-ориентированной кластеризации ПД использовалась целевая функция следующего вида:

$$\hat{f}_i = 1 - \frac{\max\left(\sum_{i=1}^{M} \overline{K}_r^i, \sum_{i=1}^{M} \hat{K}_r^i\right)}{N},$$

где $\overline{K}^i$ – множество отсутствующих документов, входящих в i-й кластер согласно сопоставлению результатов экспертного и автоматического разбиений;

$\hat{K}^i$ – множество «лишних» документов, входящих в i-й кластер согласно сопоставлению результатов экспертного и автоматического разбиений;

$i = \overline{1, M}$ – номер кластера;

M – количество кластеров;

N – количество документов.

На основе тестового множества ПД, с помощью подсистемы онтологически-ориентированного индексирования, был построен набор индексов, включающих в себя онтологические представления (ОП), оптимизированные онтологические представления (ООП) и классические индексы (КИ), содержащие пары «термин, частота». Оценка

качества кластеризации представлена в таблице 1.

Таблица 1 – Оценка качества кластеризации

Вид экспертного разбиения	Тип индекса	Время кластеризации	Значение оценочной функции
Класс документации	ОП	14,181	0,61
	ООП	27,410	**0,82**
	КИ	197,815	0,51
Вид документации	ОП	465,137	**0,29**
	ООП	1077,836	0,24
	КИ	48488,873	0,21
Раздел документации	ОП	465,137	0,23
	ООП	1077,836	**0,25**
	КИ	48488,873	0,17
Тематика работ	ОП	2273,952	**0,27**
	ООП	5315,631	0,24
	КИ	133262,323	0,14

Как видно из таблицы 1, процесс кластеризации ОП и ООП проходит быстрее (до 104 раз) относительно времени кластеризации КИ, а качество кластеризации данных типов индексов выше по сравнению с результатами кластеризации классических индексов. Временные затраты на кластеризацию ООП примерно в 2 раза больше, чем на кластеризацию ОП, и при этом ООП показывают лучшие результаты при кластеризации по классу и разделу документации.

Заключение

Результаты исследований показали, что онтологически-ориентированная кластеризация ПД является эффективной по сравнению с методом кластеризации, где документы представляются в виде набора «термин − частота». Значительно (приблизительно до 104 раз) сокращается время кластеризации онтологических представлений документов.

Библиографический список

[Берштейн и др., 2005] Берштейн А.С., Боженюк А.В. Нечеткие графы и гиперграфы. – М.: Научный мир, 2005.

[Добров и др., 2006] Добров Б.В., Лукашевич Н.В., Лингвистическая онтология по естественным наукам и технологиям: основные принципы разработки и текущее состояние // Десятая национальная конференция по искусственному интеллекту с международным участием (Обнинск, 25-28 сентября 2006 г.) – М.: Физматлит, 2006.

[Загоруйко 1999] Загоруйко Н.Г. Прикладные методы анализа данных и знаний - Новосибирск: ИМ СО РАН, 1999. - 270 с.

[Наместников 2009] Интеллектуальные проектные репозитории. – Ульяновск: УлГТУ, 2009. Наместников А.М. Интеллектуальные проектные репозитории. – Ульяновск: УлГТУ, 2009.

[Наместников и др., 2010] Наместников А.М., Филиппов А.А. Концептуальная индексация проектных документов // Автоматизация процессов управления. – 2010. – №2(20). – С. 34-39.

[Скурихин и др., 1995] Скурихин А.Н. Генетические алгоритмы // Новости искусственного интеллекта; 1995. – № 4. – С. 6–17.

ONTOLOGICAL SYSTEM FOR CLUSTERING AND FULL-TEXT SEARCHING OF THE CAD DOCUMENTS

Namestnikov A.M. *, Subkhangulov R.A. *, Filippov A.A. *

** Ulyanovsk State Technical University, Ulyanovsk, Russia*

nam@ulstu.ru

subkhangulov-ruslan@yandex.ru

al.filippov@ulstu.ru

In article the description of program system for clustering of CAD text documents is provided. The method based on a current status of domain ontology. The system architecture, structure of ontology and its fragments in the RDF format is shown, basic functions of system are listed. The results of the experiments proving efficiency of developed program system are given.

Introduction

In CAD archive the designer has no possibility to solve semi-structured problems of search. Full-text query search for the document, finding of the close document according to the contents, a clustering of documents can be such tasks.

Main Part

The architecture of the developed system is provided. The domain ontology includes two levels: conceptual and terminological. In the editor of ontology the following modules are implemented: module of formation for the diagram of ontology and module of formation for the domain maintenance; module of formation for a terminological surround of ontology concepts; the interaction module with Sesame RDF storage; the module for visualization of domain ontology.

Conclusion

Results of researches showed that method of ontological clustering is more effective than a method where documents are represented in the form of a set «term − frequency». Considerably time of a ontological clustering is reduced.