УДК 681.3

РАЗРАБОТКА ИНСТРУМЕНТАРИЯ ДЛЯ ИНТЕЛЛЕКТУАЛЬНОГО АНАЛИЗА ТЕХНИЧЕСКОЙ ДОКУМЕНТАЦИИ

© 2011 А.М. Наместников, А.А. Филиппов, Р.А. Субхангулов

Ульяновский государственный технический университет

Поступила в редакцию 21.11.2011

Данная статья является результатом исследования возможности применения онтологического подхода к индексированию и кластеризации технических документов машиностроительной отрасли. В работе рассмотрен процесс создания предметно-ориентированной онтологии, модели концептуального индекса проектного документа и модифицированного fcm-метода кластеризации концептуальных индексов.

Ключевые слова: *онтология, концептуальный индекс, кластеризация, проектный документ*

Развитие проектных репозиториев большинства машиностроительных предприятий достигло такого состояния, на котором анализ электронного архива технических документов становится весьма затруднительным. Возникает потребность в поиске новых способов хранения, систематизации и обработки текстовой информации в рамках предприятия. Это приводит к появлению новых и развитию существующих технологий. Примером является научное направление Semantic Web, в котором гипертекстовые страницы снабжаются дополнительной разметкой, несущей информацию о семантике включаемых в страницы элементов. Компонентом Semantic Web является понятие онтологии, описывающее смысл семантической разметки. На основе онтологии можно более эффективно решать задачи в области информационного поиска проектных документов, предметная область которых определена и формализована. В данной статье представлены модели, методы и инструменты, позволяющие создавать предметно-ориентированные онтологии, выполнять интеллектуальное индексирование проектных документов и их кластеризацию на основе онтологии.

Построение концептуального индекса проектного документа. Представление текстовых документов как простого набора слов имеет определенные недостатки, затрудняющие кластеризацию и информационный поиск. К ним относятся:

- избыточность — в пословном индексе используются слова-синонимы, выражающие одни и те же понятия;
- слова текста считаются независимыми друг от друга, что не соответствует словам связанного текста;
- многозначность слов – поскольку многозначные слова могут рассматриваться как дизъюнкция двух или более понятий, выражающих различные значения многозначного слова, то маловероятно, что все элементы этой дизъюнкции интересуют пользователя.

Этих недостатков лишено так называемое *концептуальное индексирование,* то есть такое индексирование, когда текст индексируется не по словам [1], а по понятиям, которые обсуждаются в данном тексте. При такой технологии

- все синонимы сведены к одному и тому же понятию,
- многозначные слова отнесены к разным понятиям,
- связи между понятиями и соответствующими словами (терминами) описаны и могут быть использованы при анализе документа.

Результатом концептуального индексирования проектного документа (ПД) в интеллектуальном проектном репозитории (ИПР) машиностроительного предприятия будем считать такое описание ПД, которое состоит из множества понятий с соответствующими степенями выраженности данных понятий в документе. Исходный ПД d поступает на вход анализатора структуры проектного документа, который на основе данных структурного уровня онтологии интеллектуального проектного репозитория выделяет отдельные структурные единицы, которые являются неделимыми (разделы, не имеющие подразделов), s_j^d, j — раздел ПД d. По каждому такому разделу

Наместников Алексей Михайлович, кандидат технических наук, доцент кафедры «Информационные системы». E-mail: nam@ulstu.ru
Филиппов Алексей Александрович, аспирант. E-mail: al.filippov@ulstu.ru
Субхангулов Руслан Айратович, аспирант

концептуальный индексатор формирует свой индекс на основе состава понятий и терминов, связанных с понятиями в онтологии ИПР. Основу концептуального индексатора составляет следующая функция:

$$F_{CI} : s_j^d \to cI_j^d \qquad (1)$$

где cI_j^d — концептуальный индекс *j*-го раздела ПД *d*.

Концептуальный индекс каждого раздела ПД поступает на вход синтезатора концептуального индекса проектного документа. Аналитически он может быть представлен следующей функцией преобразования [3]:

$$F_{SYN} : \{cI_j^d\} \to cI^d \qquad (2)$$

где cI^d — концептуальный индекс ПД *d*.

Непосредственно модель концептуальной индексации будем представлять с использованием теории графов. В онтологии ИПР каждый термин $w_i \in W$ связан с понятиями $c_j \in C$ отношением ассоциации $R_A^a : w_i R_A^a c_j$. Понятия *C* связаны друг с другом отношениями обобщения, образуя таксономию понятий предметной области. Понятийный уровень онтологии интеллектуального проектного репозитория можно представить в виде ориентированного графа:

$$G = (C, E) \qquad (3)$$

где *C* – множество вершин графа, каждая вершина – это понятие онтологии; *E* – множество дуг вида $E = \{\langle c_i, c_k \rangle\}$, для всех $c_i, c_k \in C$, для которых имеет место отношение $c_i R_G c_k$. Реализацию функции концептуального индексатора будем представлять в виде следующего алгоритма:

Шаг 1. Вычисление степеней выраженности понятий в разделе документа. Каждое отношение ассоциации между термином w_i и понятием c_j имеет вес, который характеризует частоту встречаемости термина w_i в описании понятия c_j. Такой вес определяется в процессе формирования текстового входа каждого из понятий ИПР для каждого отношения ассоциации между термином и понятием. Терминологическая составляющая *j*-го раздела ПД записывается в виде множества пар «термин-частота»:

$$\{(w_{1j}^d, f_1^j), (w_{2j}^d, f_2^j), \ldots, (w_{l_j j}^d, f_{l_j}^j)\},$$

где l_j — количество выделяемых терминов в *j*-м разделе ПД.

Текстовый вход W^k понятия c_k представим следующим образом:

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

где l_k — количество терминов текстового входа *k*-го понятия.

Степень выраженности понятия c_k в *j*-м разделе ПД *d* будем вычислять по следующей формуле:

$$\mu_{s_j^d}(c_k) = 1 - \frac{l}{l_k} \sum_{s=1}^{l_k} \left| f_s^k - f_s^j \right| \qquad (4)$$

где s_j^d - *j*-й раздел проектного документа *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*, тогда f_s^j принимается равным нулю.

Шаг 2. Определение значимых понятий. Степень выраженности каждого понятия, вычисленная по формуле 4, сравнивается с пороговым значением параметра $\eta = [0,1]$. Если $\mu_{s_j^d}(c_k) \geq \eta$, тогда понятие c_k включается в состав концептуального индекса раздела s_j^d со своей степенью выраженности.

Шаг 3. Формирование концептуального индекса. Зная состав значимых понятий со степенями выраженности, концептуальный индекс *j*-го раздела ПД *d* формируется как нечеткий граф следующего вида:

$$cI_j^d = (\tilde{C}, E) \qquad (5)$$

где $\tilde{C} = \left\{ \left\langle u_{s_j^d}(c)/c \right\rangle, c \in \{c_k : u_{s_j^d}(c_k) \geq n\} \right\}$,
$E = \{\langle c_i, c_k \rangle\}, \langle c_i, c_k \rangle \in C^2$

Очевидно, что концептуальный индекс cI_j^d является вершинным подграфом графа *G*, определяемого выражением 3. Поскольку в общем случае в концептуальном индексе могут

отсутствовать любые понятия из состава понятий онтологии ИПР, результирующий граф концептуального индекса cI_j^d может состоять из несвязанных деревьев и/или изолированных понятий.

Функция синтезатора концептуального индекса ПД F_{SYN} (2) состоит в объединении отдельных концептуальных индексов, соответствующих разделам ПД. Такое объединение выполняется поэтапно «снизу-вверх» и соответствует структуре ПД. Иерархия разделов и подразделов ПД зафиксирована в онтологии ИПР на структурном уровне (S). Поскольку в процессе интеллектуального анализа содержимого проектного репозитория может потребоваться рассмотрение не только отдельно взятых ПД, но и их разделов (например, произвести сравнительную оценку функциональных требований из разных технических заданий), синтез концептуального индекса ПД выполняется по следующему алгоритму:

1. Объединение концептуальных индексов разделов, соответствующих листовым вершинам самого нижнего уровня дерева структуры ПД.
2. Если не достигнута корневая вершина дерева структуры ПД, то объединение концептуальных индексов выполняется на текущем уровне иерархии и осуществляется переход к п.3, иначе – к п.4.
3. Перейти на шаг выше по иерархии ПД и выполнить п.2.
4. Произвести объединение концептуальных индексов, полученных на предыдущем шаге алгоритма, сформировав таким образом концептуальный индекс ПД, и остановить процесс синтеза.

Пусть cI_i^d и cI_j^d - концептуальные индексы *i*-го и *j*-го разделов ПД *d* соответственно. Объединение cI_i^d и cI_j^d определим как объединение нечетких графов:

$$cI_i^d \cup cI_j^d = (\widetilde{C}_i, E_i) \cup (\widetilde{C}_j, E_j) = (\widetilde{C}_i \cup \widetilde{C}_j, E_i \cup E_j)$$

причем

$$\widetilde{C}_i \cup \widetilde{C}_j = \{\mu_{s_i^d}(c) \vee \mu_{s_j^d}(c)/c\},$$

$$c \in \{c_k : \mu_{s_i^d}(c_k) \geq \eta\} \cup \{c_m : \mu_{s_j^d}(c_m) \geq \eta\}.$$

Другими словами, объединение концептуальных индексов есть объединение вершин и дуг нечетких графов, которые их представляют. При этом результирующая степень выраженности понятия есть дизъюнкция исходных степеней. В частности, для нечеткой интерпретации можно записать следующим образом:

$$\widetilde{C}_i \cup \widetilde{C}_j = \{\max(\mu_{s_i^d}(c) \vee \mu_{s_j^d}(c))/c\},$$

$$c \in \{c_k : \mu_{s_i^d}(c_k) \geq \eta\} \cup \{c_m : \mu_{s_j^d}(c_m) \geq \eta\}.$$

Таким образом, получаем, что ПД в ИПР представляется не в лексическом пространстве терминов, которые удается выделить в документе, а в пространстве понятий предметной области, которые зафиксированы в онтологии ИПР.

Модифицированный метод кластеризации концептуальных индексов. Fuzzy c-means (FCM) является методом кластеризации, который позволяет одному объекту принадлежать двум или более кластерам с определенной степенью принадлежности. Этот метод (разработанный J.C. Dunn в 1973 г. и улучшенный J.C. Bezdek в 1981 г.) часто используется при решении задачи распознавания образов. Алгоритм основан на минимизации следующей целевой функции:

$$J_m = \sum_{i=1}^{N} \sum_{j=1}^{C} u_{ij}^m \left\| cI_i - cI_j^c \right\|^2, 1 \leq m < \infty,$$

где N – количество концептуальных индексов для кластеризации, C – количество кластеров, m – любое действительное число больше 1, u_{ij} – степень принадлежности концептуального индекса cI_i кластеру j, cI_i – i-ый концептуальный индекс, cI_j^c – центр j-го кластера, ||*|| – нормализованное расстояние между концептуальным индексом и центром кластера. Так как структура для всего набора концептуальных индексов одинакова (в случае нахождения меры сходства содержимого – являются нечеткими вершинными подграфами одного и того же графа – онтологии ИПР, в случае нахождения меры сходства структуры – добавляются недостающие вершины), будем рассматривать нечёткий граф концептуального индекса, как вектор, содержащий значения понятий и их степеней выраженности, либо метки разделов со значениями структурных индексов данных разделов.

FCM алгоритм состоит из следующих шагов:

Шаг 1. Инициализация. Задаются параметры кластеризации и инициализируется первоначальная матрица принадлежности концептуальных индексов кластерам $U = \lfloor u_{ij} \rfloor$.

Шаг 2. Вычисление центров кластеров.

Вычисляется новое значение центров кластеров:

$$cI_j^c = \frac{\sum_{i=1}^{N} u_{ij}^m * cI_i}{\sum_{i=1}^{N} u_{ij}^m},$$

Шаг 3. Формирование новой матрицы принадлежности. Формируется новая матрица принадлежности с учетом вычисленных на предыдущем шаге центров кластеров:

$$u_{ij} = \frac{1}{\sum_{l=1}^{C} \left(\frac{\left\| cI_i - cI_j^c \right\|}{\left\| cI_i - cI_l^c \right\|} \right)^{\frac{2}{m-1}}},$$

где u_{ij} – степень принадлежности *i*-го концептуального индекса кластеру *j*, cI_j^c – концептуальный индекс центра *j*-го кластера, cI_l^c – концептуальный индекс центра *l*-го кластера.

Шаг 4. Вычисление целевой функции. Вычисляется значение целевой функции, и полученное значение сравнивается со значением на предыдущей итерации. Если разность не превышает заданного в параметрах кластеризации порогового значения, считаем, что кластеризация завершена. В противном случае переходим ко второму шагу алгоритма. Для определения расстояния между содержимым ПД в ИПР необходимо измерить степень близости, похожести между концептуальными индексами ПД. В рамках данного исследования можно выделить две меры сходства ПД: мера сходства содержимого ПД; мера сходства структур ПД.

Будем рассматривать концептуальный индекс ПД как иерархию. Тем самым, расстояние между содержимым ПД находится через сложность превращения одной иерархии в другую, путем вычисления разности между степенями выраженности понятий, имеющих одинаковые метки (имя) [2]. Рассмотрим определение понятия *иерархия*, представленное в работе [2]. Обозначим через W конечное множество объектов, $W = w_1, w_2, ..., w_l, ..., w_q$, а через H – множество непустых частей множества W, называемых *таксонами* и обозначаемых через h. *Иерархией H множества W* называется множество подмножеств W таких, что:

- $\forall w \in W \{w\} \in H$ (терминальные вершины (листья) – одноэлементные множества);
- $W \in H$ (наибольший таксон (корень) содержит все элементы W);
- для любых вершин $h, h' \in H$ мы имеем либо $h \cap h' = 0$, либо $h \subset h'$, либо $h' \subset h$.

Таким образом, иерархия – это многоуровневая структура, в которой объекты, находящиеся в одном таксоне на некотором (*j*-м) уровне, остаются в одном таксоне на (*j*+1)-м и всех других более высоких уровнях. Первому уровню соответствуют терминальные вершины (п. 1 в определении иерархии), а последнему, максимальному, уровню обозначим его через *m*) – наибольший таксон, содержащий все элементы W; этот таксон можно обозначить тем же символом W (п. 2 в определении иерархии). На каждом уровне происходит или не происходит объединение таксонов (п. 3 в определении иерархии).

Практическая реализация интеллектуального проектного репозитория. Для решения задачи кластеризации концептуальных индексов проектных документов была выполнена работа над программной реализацией ИПР. Языком разработки был выбран язык Java. В качестве хранилища исходных и проиндексированных документов используется XML-ориентированная СУБД Tamino (Software AG), доступ к которой осуществляется с помощью Tamino API. В качестве хранилища онтологий используется Java фреймворк Sesame в связке с веб-сервером Apache Tomcat. Структура приложения представлена на рис. 1.

Для построения онтологии интеллектуального проектного репозитория было выбрано хранилище Sesame − открытая (open source) база данных RDF с поддержкой логического вывода по RDF-тройкам и запросов. Оно предлагает большой набор инструментов для разработчиков для использования RDF и RDF Schema. Рассмотрим пример части онтологии ИПР. В онтологии представлены описания понятий, терминов, из которых состоят понятия, и их отношения. В рассматриваемой онтологии

- понятия описываются с помощью класса <rdfs:Class rdf:ID="Concept"/>,
- термы с помощью класса <rdfs:Class rdf:ID="Term"/>,
- концепт-термы (объекты, представляющие отношения между понятиями и термами) с помощью класса <rdfs:Class rdf:ID="ConceptTerm"/>,
- отношения между понятиями с помощью свойства

<rdf:Property rdf:ID="IsASubconcept">
<rdfs:domain rdf:resource="#Concept"/>
<rdfs:range rdf:resource="#Concept"/>
</rdf:Property>,

- отношения между концепт-термами и понятиями с помощью свойства
<rdf:Property rdf:ID="AssociatedWithConcept">
<rdfs:domain rdf:resource="#ConceptTerm"/>
<rdfs:range rdf:resource="#Concept"/>
</rdf:Property>,
- отношения между концепт-термами и термами с помощью свойства
<rdf:Property rdf:ID="AssociatedWithTerm">
<rdfs:domain rdf:resource="#ConceptTerm"/>
<rdfs:range rdf:resource="#Term"/>
</rdf:Property>,
- относительная частота встречаемости терма в понятии с помощью свойства
<rdf:Property rdf:ID="HasAFreq">
<rdfs:domain rdf:resource="#ConceptTerm"/>
<rdfs:range rdf:resource="&xsd;float"/>
</rdf:Property>.

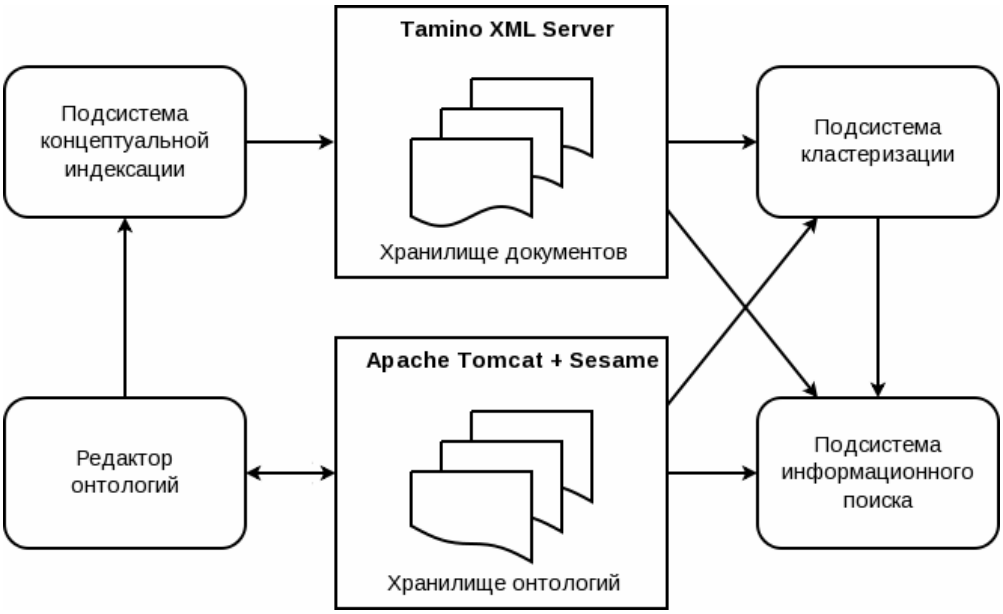

Рис. 1. Архитектура ИПР

Этапы индексации проектных документов:

- загрузка документов;
- анализ структуры документов;
- удаление стоп-слов;
- стемминг;
- подсчёт относительной частоты встречаемости термов;
- расчёт степени выраженности понятий;
- построение концептуальных индексов для разделов и документов.

Для работы с xml документами используется SAX-парсер («Simple API for XML») из стандартной поставки Java. Для реализации функции стемминга используется стеммер Snowball. Стемминг – это процесс нахождения основы слова для заданного исходного слова. В результате работы механизма стемминга получается документ, состоящий из термов. Под термом стоит понимать лексическую единицу, полученную в результате процесса стемминга.

Для работы с графом был разработан класс JGraph. Методы данного класса позволяют добавлять и удалять вершины из графа, а также устанавливать степень выраженности каждой вершине. Также существует возможность сохранять и загружать граф из XML файла. Ниже представлен пример XML файла, содержащего граф:

```
<?xml version="1.0" encoding="windows-1251" ?>
<graph>
<vertex name="система" value="0.5" parent="ROOT" />
<vertex name="информационная система" value="0.7" parent="система" />
<vertex name="техническая система" value="0.3" parent="система" />
<vertex name="система управления базами данных"
value="0.1" parent="информационная система" />
</graph> .
```

Ниже представлены основные экранные формы подсистемы концептуальной индексации ПД (рис. 2, 3) и подсистемы нечеткой кластеризации концептуальных индексов ПД (рис. 4).

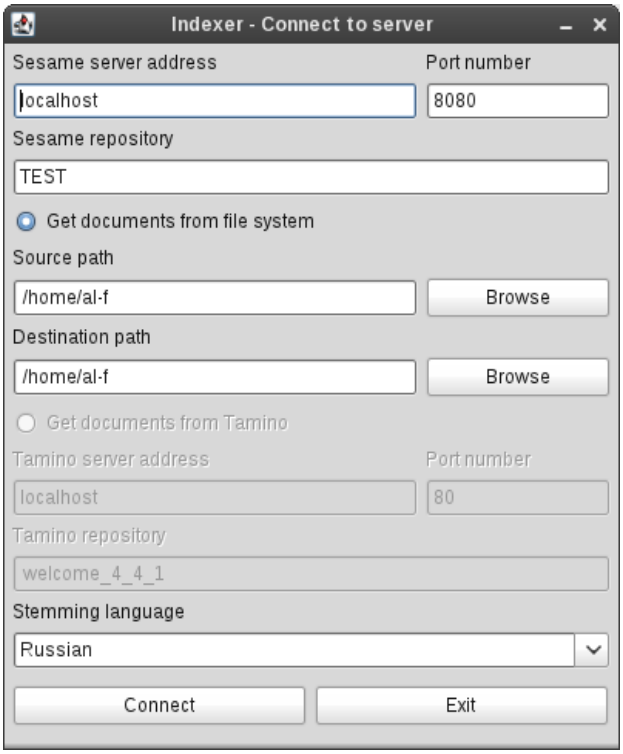

Рис. 2. Диалоговое окно настройки подсистемы концептуальной индексации

Выводы: разработанные модели и методы нечеткого индексирования и кластеризации проектных документов машиностроительного предприятия позволяют использовать онтологическое описание предметной области в качестве средства управления процессом интеллектуального анализа проектных документов. Результаты концептуального индексирования и кластеризации проектных документов определяются тем набором понятий предметной области, который представлен в онтологии. Экспериментальные исследования с разработанными программными модулями редактирования онтологии, концептуального индексирования и кластеризации подтверждают адекватность представленных в статье теоретических моделей.

СПИСОК ЛИТЕРАТУРЫ:

1. *Наместников, А.М.* Интеллектуальный сетевой архив электронных информационных ресурсов / *А.М. Наместников, А.В. Чекина, Н.В. Корунова* // Программные продукты и системы. 2007. № 4. С. 10-13.
2. *Загоруйко, Н.Г.* Прикладные методы анализа данных и знаний. – Новосибирск: ИМ СО РАН, 1999. 270 с.
3. *Наместников, А.М.* Концептуальная индексация проектных документов / *А.М. Наместников, А.А. Филиппов* //Автоматизация процессов управления. 2010. №2(20). С. 34-39.

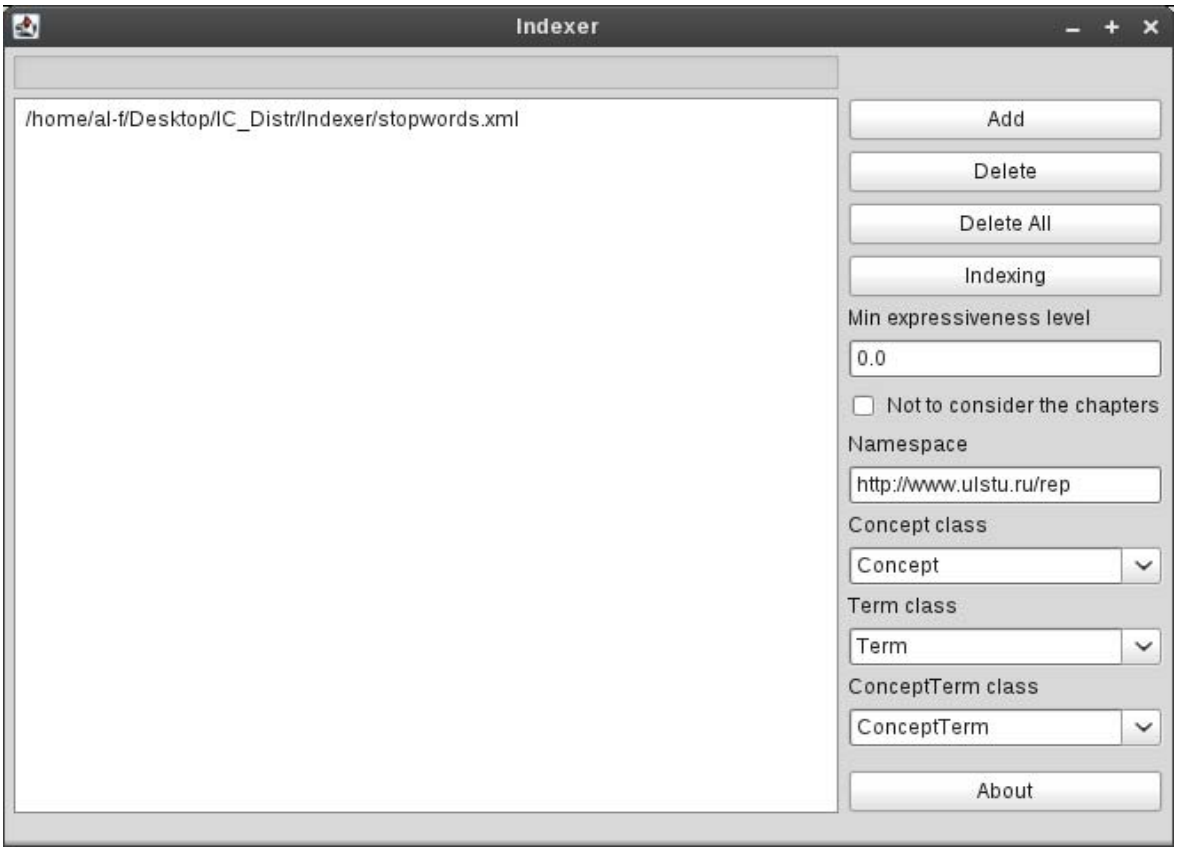

Рис. 3. Основная экранная форма подсистемы концептуальной индексации

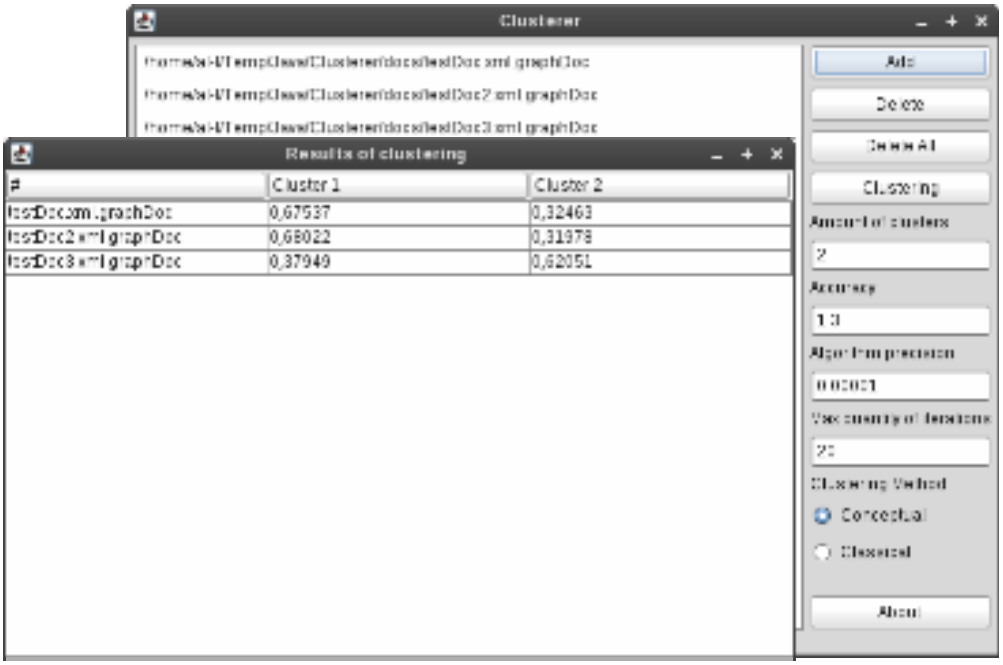

Рис. 4. Иллюстративный пример вывода результата нечеткой кластеризации концептуальных индексов

DEVELOPMENT OF TOOLKIT FOR THE INTELLECTUAL ANALYSIS OF ENGINEERING SPECIFICATIONS

© 2011 A.M. Namestnikov, A.A. Filippov, R.A. Subhangulov

Ulyanovsk State Technical University

Given article is a result of researching the possibility of application the onthologic approach to codeindexing and clusterization of technical documentation in machine-building branch. In work the process of creation the subject-oriented onthology, models of conceptual index of design document and the updated fcm-method in clusterization the conceptual indexes is considered.

Key words: *onthology, conceptual index, clusterization, design document*

Aleksey Manestnikov, Candidate of Technical Sciences, Associate Professor at the Intelligence Systems Department. E-mail: nam@ulstu.ru
Aleksey Filippov, Post-graduate Student. E-mail: al.filippov@ulstu.ru
Ruslan Subhangulov, Post-graduate Student