Применение концептуальных графов в системах поддержки электронных библиотек♣

© Богатырев М.Ю., Латов В.Е., Столбовская И.А.

Тульский государственный университет
okkambo@mail.ru

Аннотация

Работа содержит некоторые результаты исследований возможностей применения семантических моделей текста в виде концептуальных графов в качестве объектов хранения электронных библиотек. Рассматриваются постановки и решения задач кластеризации концептуальных графов.

1 Введение

Одним из направлений развития современных электронных библиотек является расширение их функциональных возможностей. Такое расширение было бы весьма значительным, если бы в библиотеках хранились не только тексты, но и их смысловое содержание. Решить эту задачу можно, исследуя и реализуя семантические модели текстов.

В данной работе предлагается применить одну из семантических моделей текста - *концептуальный граф* [8] – в качестве объекта хранения электронной библиотеки.

Применение концептуальных графов позволяет развивать технологии поддержки электронных библиотек, по крайней мере, в двух направлениях:

- автоматизация построения каталогов библиотек, модификация и коррекция существующих каталогов на основе анализа потока входных текстов;
- извлечение знаний из электронных библиотек в виде *концепций* и *онтологий*.

Оба указанных направления в настоящее время представлены множеством методов и технологий [2, 3]. Эффективное применение здесь концептуальных графов связано с решением задач *агрегирования* и *кластеризации* на графах.

Концептуальные графы применяются в разрабатываемой исследовательской системе – прототипе электронной библиотеки. Библиотека содержит научные статьи вместе с аннотациями. Концептуальные графы строятся для аннотаций статей, поскольку аннотации статей призваны сжато и точно отражать их содержание. Поэтому применение аннотаций вместо текста статьи обеспечивает необходимую информативность и компактность.

2 Концептуальные графы и их особенности

Концептуальный граф признан в качестве одной из семантических моделей, применяемых для анализа текстов. Вместе с *концептуальными решетками* концептуальные графы относятся к *концептуальным структурам*, которые являются одним из формальных представлений знаний [7].

Концептуальный граф - это двудольный направленный граф, состоящий из двух типов узлов: *концептов* и *концептуальных отношений*. На рис.1 показан пример концептуального графа для предложения: «Студент читает книгу».

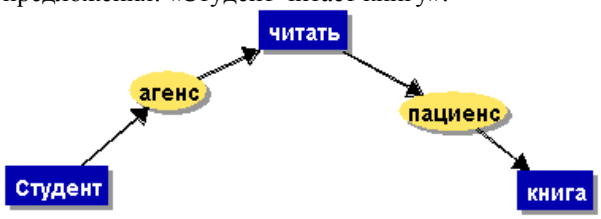

Рис. 1. Пример концептуального графа. Прямоугольники обозначают концепты, эллипсы – отношения.

2.1 Форматы концептуальных графов

Для концептуальных графов разработан стандарт их представления [8] и языки описания, среди которых наиболее популярны CGIF (Conceptual Graph Interchange Form) и XML - представление концептуальных графов.

В формате CGIF концептуальный граф на рис.1 записывается следующим образом:

[Студент*a:"] [Читать*b:"] [Книга*c:"] (Пациенс?b?c) (Агенс?a?b).

Здесь a, b, c – метки, присваемые концептам.

Так называемая линейная форма представления концептуальных графов более удобна для задания их в тексте:

[Студент] → (Агенс) → [Читать] → (Пациенс) → [Книга]

Приведенная здесь краткая линейная форма может быть расширена, например:

[Студент:“ Иванов”]→ …

Ниже приведен фрагмент XML – представления того же графа:

```
<?xml version="1.0" encoding="UTF-8"?>
<conceptualgraph editor="CharGer" version="3.5b1" created="17.07.2007 11:55:40" user="Elbereth">
  <graph id="128173821741" owner="0">
    <type>
      <label>Proposition</label>
    </type>
    <layout>
      <rectangle x="5.0" y="5.0" width="900.0" height="750.0"/>
      <color foreground="0,0,175" background="0,0,175"/>
    </layout>
    <concept id="128173821742" owner="128173821741">
      <type>
        <label>Студент</label>
      </type>
      <layout>
            …..
```

Отметим возможность при помощи XML не только задавать собственно граф, но и указывать элементы и параметры его визуализации, что видно из приведенного примера кода XML.

Поддержка концептуальных графов в электронных библиотеках означает решение следующих задач:

- построение концептуальных графов;
- организацию хранения концептуальных графов в выбранном формате;
- решение прикладных задач через реализацию соответствующих алгоритмов, использующих операции на концептуальных графах.

2.2 Построение и хранение концептуальных графов

В настоящее время нет систем, позволяющих автоматически строить концептуальный граф по предъявленному тексту на русском языке. Подобные англоязычные системы также далеки от совершенства [2].

Поскольку концептуальный граф задает смысл порождающего его предложения, он не может быть построен однозначно. Выбор концептов и отношений субъективен, но выполняется в рамках известных лингвистических моделей и подходов к анализу текста. Естественными лингвистическими подходами к построению концептуальных графов являются *грамматический* (концепты – глаголы и существительные) и *семантический* (применение *семантических ролей* в отношениях). Так в примере на рис.1 в качестве концептуальных отношений непосредственно использованы известные в лингвистике семантические роли: *агенс* и *пациенс* [7].

Автоматизация построения концептуальных графов является нерешенной проблемой и широко обсуждается (см., например, [1]), однако, выходит за рамки данной работы.

На практике широко применяется полуавтоматический способ построения концептуальных графов при помощи программ-редакторов.

Принципиальную возможность применения концептуальных графов к решению упомянутых выше проблем автоматизации построения каталогов и извлечения знаний проиллюстрируем следующим образом.

1. Каждый концептуальный граф задает смысл порождающего его предложения. Другими словами, концептуальный граф отвечает на вопрос, о чем данное предложение. Для этого в концептуальных графах вводят понятие *центральный концепт*. Таких концептов может быть несколько. Для графа на рис.1 центральным концептом является глагол «читать». Следовательно, предложение «Студент читает книгу» - о чтении.

Очевидно, что множество предложений предъявленного текста можно разбить на подмножества предложений, объединенных одной темой, хотя среди этих подмножеств будут и подмножества, включающие лишь одно предложение. Например, предложения, которым соответствуют концептуальные графы с центральным концептом «читать», образуют тему «чтение». Такие подмножества – кластеры задают тематику данного текста. Всякая тематика иерархична: например, в теме «чтение» существуют подчиненные ей варианты, характеризующие особенности чтения – внимательное, вслух и т.д., которые находят воплощение в соответствующих концептуальных графах.

Таким образом, приходим к задаче иерархической кластеризации на концептуальных графах. Объединение графов в кластеры соответствует определению *тематических групп*, присутствующих в тексте. Центральные концепты в иерархиях кластеров задают темы, которые могут

Труды 9ой Всероссийской научной конференции «Электронные библиотеки: перспективные методы и технологии, электронные коллекции» - RCDL’2007, Переславль-Залесский, Россия, 2007.

составить ветви дерева каталога, соответствующего данному тексту.

2. Извлечение знаний из текстов (в более широкой трактовке - Text Mining) представляет собой обширное направление исследований, предлагающее множество методов построения и применения моделей знаний [12]. Знание, извлеченное из текста, аналогично предыдущему случаю, представляет собой ответ на вопрос: «О чем данный текст?» Решение строится в рамках некоторой модели знаний, например, в виде *онтологии*.

Онтология как структура, объединяющая понятия (концепции) представляется концептуальным графом, построенным агрегированием некоторого множества концептуальных графов. В качестве множеств - кандидатов на агрегирование вполне естественно применить кластеры, выделенные из исходного множества концептуальных графов. Каждый кластер соответствует одной теме - *концепции*. Вместе с тем, он содержит определенные детали темы, представленные как центральными, так и обычными концептами графов, составляющих кластер. Построив из концептуальных графов кластера новый агрегированный граф, получим онтологию.

Таким образом, в основе решения двух важных проблем поддержки электронных библиотек лежит решение задачи *иерархической кластеризации на концептуальных графах*.

3 Кластеризация на концептуальных графах

В любой задаче кластеризации главной проблемой является построение меры близости кластеризуемых объектов. Под мерой близости концептуальных графов понимается количественная характеристика, призванная отразить семантическую близость порождающих графы предложений, что сделать, очевидно, полностью невозможно. Поэтому проблема близости остается центральной в анализе концептуальных графов.

3.1 Меры близости концептуальных графов

Для текстовых документов одной из распространенных мер близости является коэффициент Дайса [6]:

$$S_{D_1,D_2} = \frac{2n(D_1 \cap D_2)}{n(D_1)+n(D_2)},$$

где $n(D_i)$ – число терминов в документе D_i, $n(D_1 \cap D_2)$ – число терминов, содержащихся как в документе D_1, так и в документе D_2.

Данный подход положен в основу расчета меры близости концептуальных графов. Для двух графов G_1 и G_2 мера близости зависит от двух значений: концептуальной близости s_c и относительной близости s_r.

$$s_c = \frac{2n(G_c)}{n(G_1)+n(G_2)}, \qquad (1)$$

где $G_c = G_1 \cap G_2$, $n(G)$ – число концепций – концептуальных узлов графа G.

$$s_r = \frac{2m(G_c)}{m_{G_c}(G_1)+m_{G_c}(G_2)}, \qquad (2)$$

где $m(G_c)$ – число отношений – относительных узлов концептуального графа G_c, $m_{G_c}(G)$ – число отношений – относительных узлов концептуального графа G, для которых хотя бы одна из вершин принадлежит графу G_c.

Меры близости (1), (2) несовершенны. В результате их применения близкими могут оказаться графы, имеющие просто большое число одинаковых концептов или отношений, но по смыслу совершенно далекие.

Рассмотрим пример.

Определим концептуальные близости графов, построенных для следующих предложений:

1. “Статья описывает генетический алгоритм”.
2. “ Генетический алгоритм рассматривается в данном документе”.
3. “Статья описывает концептуальный граф”.

Концептуальные графы для данных предложений имеют вид:

1. [статья] → (агенс) → [описывать] → (пациенс) → [алгоритм] → (атрибут) → [генетический]
2. [генетический] ← (атрибут) ← [алгоритм] ← (адресат) ← [рассматриваться] → (источник) → [документ] → (атрибут) → [данный]
3. [статья]→ (агенс) → [описывать] → (пациенс) → [граф] → (атрибут) → [концептуальный]

Концептуальные близости графов 1 и 2, 1 и 3 следующие:

$$s_{c1-2} = 2*2/(4+5) = \frac{4}{9};\ s_{c1-3} = 2*2/(4+4) = \frac{4}{8} = 0.5$$

Предложения 1 и 3 оказались концептуально достаточно близкими, что неверно, тогда как предложения 1 и 2, в действительности, об одном и том же, имеют несколько меньшую близость, чем предложения 1 и 3.

Данное свойство стандартных мер близости (1), (2) хорошо известно. Неправильный результат вычисления близости возникает из-за присутствия в предложениях одинаковых слов, не несущих нужной информации.

Такие слова – *клише, штампы, стандартные грамматические обороты* – присутствуют во всех языках. Для удаления подобного «шума» необходима фильтрация предложений.

В общей постановке задача семантической фильтрации достаточно сложна и ее решение

примыкает к решению задачи распознавания смысла текста.

Учитывая введенное ограничение - анализируемые тексты являются аннотациями научных статей, - удалось сформировать множество *общезначимых концептов* научной тематики, которые можно исключить при анализе близости концептуальных графов. В рассмотренном примере к таким концептам относятся: “ статья”, “данный”, “документ”, “рассматривать(ся)”, “описывать(ся)”.

С учетом общезначимости концепций в качестве величины $n(G)$ для формулы (1) принимается число концептов, не являющихся общезначимыми:

$$n(G) = a_1 + a_2 + ... + a_i \ , \ i = 1,...,N \ , \ a_i \in \{0\,|1\}\,,$$

где N – число всех концептов графа G , a_i – коэффициент общезначимости – принимает значения 0 или 1 в зависимости от того, является ли i -я концепция общезначимой или нет.

В результате фильтрации концептуальные близости графов 1 и 2, 1 и 3 вычисляется следующим образом: $s_{c1-2} = 2*2/(2+2) = 1$; $s_{c1-3} = 2*0/(2+2) = 0$, что более соответствует смыслу анализируемых предложений как фрагментов научного текста.

При расчете концептуальной близости графов необходимо также учитывать размеры сравниваемых графов. Например, концептуальная близость двух графов, содержащих соответственно 2 и 10 концептов, с 2-мя общими концептами равна $s_{c1} = 2*2/(10+2) = \frac{1}{3}$, тогда как концептуальная близость двух графов, содержащих по 3 концептов и имеющих только один общий концепт, также равна $s_{c2} = 2*1/(3+3) = \frac{1}{3}$. В первом случае граф, содержащий 2 концепта, является подграфом графа, содержащего, 10 концептов, тогда как во втором случае сравниваемые графы имеют значительно меньше общего.

Возможна модификация близости (1), (2), позволяющая учитывать размеры графов:

$$s_c = \frac{2n(G_c)l}{n(G_1) + n(G_2)}, \qquad (3)$$

где

$$l = \begin{cases} k\dfrac{n(G_1)}{n(G_2)}, если\ n(G_1) \geq n(G_2) \\ k\dfrac{n(G_2)}{n(G_1)}, если\ n(G_1) < n(G_2) \end{cases},$$

k – масштабирующий коэффициент.

При расчете относительной близости концептуальных графов с помощью формулы (2) учитывается число отношений графа.

С учетом значимости отношений величина $m_{G_c}(G)$ для формулы (2) вычисляется следующим образом:

$m_{G_c}(G) = m_{both} + b_1 + b_2 + ... + b_i$,

$i = 1,...,m - m_{both}$, $b_i \in \{0,1\}$, где m – число всех отношений графа G, m_{both} – число отношений концептуального графа G, для которых обе вершины принадлежит графу G_c, b_i – коэффициент значимости – принимает значения от 0 до 1, в зависимости от типа отношения.

Под мерой близости двух концептуальных графов, принимающей значение от 0 до 1, будем понимать значение

$$s = d_1 s_c + d_2 s_r, \qquad (4)$$

где d_i – масштабирующие коэффициенты, регулирующие баланс концептуальной и относительной близостей в мере близости s.

Применение единой меры близости предпочтительно при анализе большого числа объектов.

3.2 Алгоритмы кластеризации

Алгоритмы кластеризации на графах традиционно используют в качестве меры близости вершин численные характеристики, связываемые с дугами графа.

В данном случае речь идет о сравнении графов между собой.

Потому рассматриваемая здесь задача кластеризации концептуальных графов не есть задача кластеризации на графах. Это задача кластеризации объектов, являющихся графами.

На рис. 2 показаны результаты эксперимента по кластеризации концептуальных графов, соответствующих текстам аннотаций двух научных статей. Первая аннотация имеет 6 предложений, а вторая 4 предложения. Статьи имеют различные темы. Применялся стандартный алгоритм дивизимного типа [13] из пакета ClusterAnalysis, вычислительной системы *Mathematica* [10].

Результаты изображены в виде дендограммы, отражающей иерархическую структуру полученных кластеров.

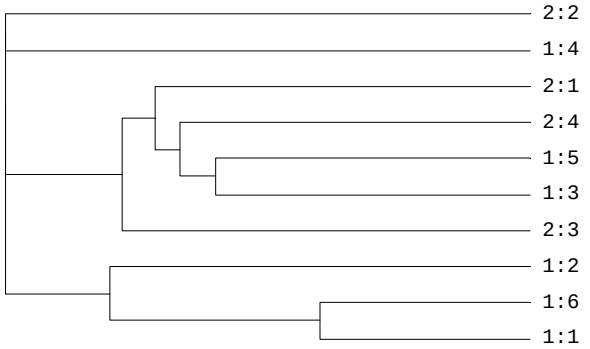

Рис. 2. Пример кластеризации концептуальных графов, соответствующих текстам аннотаций двух

научных статей. Числа справа означают номер аннотации : номер предложения в аннотации.

Как видно из дендограммы, в целом полученная кластеризация отражает различие тем аннотаций. Однако предложения 3 и 5 из первой аннотации ошибочно отнесены к иерархии кластеров, относящейся ко второй аннотации.

Причина ошибки в том, что тексты обеих аннотаций содержат одинаковые слова - концепты, не являющиеся общезначимыми, а наоборот – важными терминами. Фильтрация общезначимых концептов в данном случае бесполезна.

Данный пример приводит нас к более общей проблеме *поддержки контекстов*.

Контекст как семантическое понятие имеет приближение в языке концептуальных графов.

Контекстом называется концепт, которому может быть сопоставлен хотя бы один непустой концептуальный граф [8].

Другими словами, наличие контекста у некоторого предложения, которому соответствует концептуальный граф, означает, что данному предложению можно поставить в соответствие одно понятие – концепт. Этот концепт не принадлежит концептуальному графу анализируемого предложения и может входить в другой концептуальный граф (изолированные концепты также допускаются).

Так предложение «В статье исследуется генный состав хромосом» может относиться как к биологии (генетика), так и к прикладной математике (методы оптимизации, генетические алгоритмы).

Два концептуальных графа, содержащих слова из данного предложения или их производные, окажутся близкими. В результате различие контекстов будет потеряно.

Следовательно, необходимо сравнивать концептуальные графы не только по их внутреннему строению (меры (1) - (4)), но и по принадлежности контекстам.

Такая задача актуальна для электронных библиотек, когда вновь поступающий объект хранения должен быть классифицирован в соответствие с каталогом. Строить каталоги электронных библиотек можно также с использованием концептуальных графов.

Очевидно, что в этом случае кроме мер близости (1) - (4) необходимо применять меры близости концептуальных графов, измеряемых на контекстах.

Контексты являются представителями кластеров - предметных областей, реализованных, например, в тезаурусе библиотеки. Следовательно, в качестве первого приближения меры близости концептуального графа контексту можно выбрать некоторую метку кластера – контекста, например, его номер.

Здесь мы приходим к другой задаче кластеризации концептуальных графов, которую в отличие от предыдущей, *локальной* задачи, можно условно назвать задачей *глобальной кластеризации*.

Суть в том, что здесь близость концептуальных графов определяется *вне самих концептуальных графов* – через принадлежность их определенным концептам.

Если представить меры близости концептуальных графов как функции, определенные на предметных областях, то такие функции, очевидно, существенно разрывны. С другой стороны, что характерно для концептуальных графов, конфигурации кластеров и их границы могут быть достаточно размыты.

Как известно [5], в таких случаях стандартные методы кластеризации уступают методам, способным максимально исследовать пространство поиска решений, применяя, например, случайный поиск.

3.3 Эволюционный подход к кластеризации

Решение всякой задачи кластеризации основано на нахождении экстремума функции близости кластеризуемых объектов, то есть, связано с решением задачи оптимизации.

Для решения рассмотренной выше задачи глобальной кластеризации с учетом особенностей такой задачи – прежде всего, из-за разрывности целевой функции - исследовался подход, основанный на методах эволюционных вычислений [11], использующих генетические алгоритмы в задачах кластеризации [9], [5].

Эволюционные вычисления составляют класс методов оптимизации, эффективных для слабо формализованных задач оптимизации, а также для задач, оптимизируемые функции в которых мультимодальны и имеют разрывы. Основу эволюционных вычислений составляют *генетические алгоритмы*.

Эволюционный алгоритм кластеризации концептуальных графов использует *кодирование решений* в заданном алфавите и работает с *популяцией решений* согласно схемам генетических алгоритмов [9].

Принципиальное значение при использовании генетических алгоритмов имеет способ кодирования решений в виде строк – хромосом.

Известны несколько способов кодирования решений в задачах кластеризации. Все они используют *длинные хромосомы*, в которых длина совпадает с числом объектов кластеризации, а позиции строки – гены – задают определенным способом вариант кластеризации.

Например, такая хромосома имеет вид: $a_1 a_2 ... a_n$, где $a_i \in \{1,...,k\}$, *k*- заданное число кластеров, *n*- число объектов кластеризации, a_i указывает кластер, в котором находится *i*-ый объект. Недостаток кодировки состоит в том, что необходимо указать число кластеров, которое обычно априори неизвестно. Кроме того, на длинных хромосомах генетические алгоритмы работают медленно.

В результате исследований была предложена модифицированная кодировка, в которой a_i указывает на порядковый номер объекта, в одном кластере с которым находится i - ый объект. В результате количество кластеров зависит от распределения таких связей между объектами. Если объект А указывает на объект В, а тот, в свою очередь, на объект С, то это означает, что все они находятся в одном кластере.

Таким образом, данная кодировка не привязывает объекты к какому-то кластеру, а содержит только информацию о связях объектов друг с другом. Благодаря «*цепочечной*» связи объектов для объединения двух совокупностей достаточно лишь добавить связь между любым объектом из первой совокупности и любым объектом из второй. Это существенно ускорило работу алгоритма.

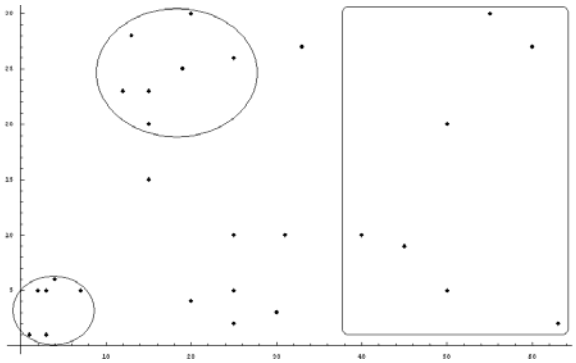

Рис. 3. Кластеризация структры концептуальных графов, принадлежащих двум предметным областям.

В результате были успешно кластеризованы структуры концептуальных графов, типичная из которых показана на рис. 3. Меры принадлежности концептуальных графов предметным областям задавались в виде шкал, размер которых определялся *представительностью* данной области; шкалы на рис. 3 имеют 25 и 75 элементов, соответственно.

4. Проект электронной библиотеки.

Результаты данного исследования применяются в пилотном некоммерческом проекте электронной библиотеки с функциями. автоматизации построения каталогов библиотеки и извлечения знаний в виде концепций и онтологий.

Для построения концептуальных графов применяется русскоязычный вариант редактора концептуальных графов, построенного с использованием Java - классов открытых программных интерфейсов Notio API [3]. Используя интерфейсы Notio, организован доступ к концептам и отношениям концептуальных графов, что позволяет выполнять операции с концептуальными графами, необходимые далее для решения задач агрегирования и кластеризации. К таким операциям относятся *сравнение двух графов*, *нахождение общих подграфов* и т.п.

Данное решение позволяет строить XML – базу данных концептуальных графов в СУБД Sybase. Выбор данной СУБД обусловлен наличием в ней эффективной технологии поддержки XML – баз данных.

Для экспериментальных исследований алгоритмов кластеризации применяется система *Mathematica*, для чего был создан интерфейс данной системы с СУБД Sybase.

Проект представляет собой удобную программную систему для выполнения вычислительных экспериментов с концептуальными графами и различными алгоритмами их обработки и кластеризации.

Главным недостатком в настоящее время является отсутствие подсистемы автоматического построения концептуальных графов, которая позволила бы реализовать проект on line.

Литература

[1]. A World of Conceptual Graphs: http://conceptualgraphs.org/

[2]. Boytcheva, S. Dobrev, P. Angelova, G.CGExtract: Towards Extraction of Conceptual Graphs from Controlled English. Lecture Notes in Computer Science № 2120, Springer 2001.

[3]. F. Southey J. G. Linders. Notio - A Java API for Developing CG Tools. 7th International Conference on Conceptual Structures, 1999. P.p. 262-271.

[4]. Hirst G. Ontology and the Lexicon. - Handbook on Ontologies in Information Systems, Berlin – Springer, 2003.

[5]. Krovi, R. Genetic algorithms for clustering: A preliminary investigation. – Proc. of the Twenty Fifth Hawaii International Conference on System Sciences. p.p. 540-544 IEEE Computer Society Press, 1991.

[6]. Montes-y-Gomez, Gelbukh, Lopez-Lopez, Baeza-Yates, Flexible Comparison of Conceptual Graphs. Lecture Notes in Computer Science 2113. Springer-Verlag, 2001.

[7]. Sarbo, J. Formal conceptual structure in language. In Dubois, D. M., editor, Proceedings of Computing Anticipatory Systems (CASYS'98), pp. 289 -300, Woodbury, New York. 1999.

[8]. Sowa R., Conceptual Graphs: Draft Proposed American National Standard, International Conference on Conceptual Structures ICCS-99, Lecture Notes in Artificial Intelligence 1640, Springer 1999.

[9]. Богатырёв М.Ю. , Латов В.Е. Исследование генетических алгоритмов кластеризации. - Изв. ТулГУ. Сер. Математика. Механика. Информатика. Том 8, вып. 3 . Информатика. - Тула, 2002. - С. 101- 107.

[10]. Богатырёв М.Ю. Прикладное моделирование в системе *Mathematica*. Основы работы с системой: Учеб. пособие по спец. 071900 «Информационные системы в технике и технологиях». – Тула, ТулГУ, 2003. – 176 с.
[11]. Богатырев М.Ю., Ковалев Д.А., Евсюков В.В. Эволюционный подход к извлечению знаний из реляционных баз данных в корпоративных информационных системах – «Информационные технологии», 2004, № 9. – С. 19-27.
[12]. Городецкий В.И., Самойлов В.В., Малов А.О. Современное состояние технологии извлечения знаний из баз и хранилищ данных. – Журнал Российской ассоц. искусственного интеллекта, 2002, № 3. – С. 3-31.
[13]. Дюран Б., Оделл П. Кластерный анализ – М.: Статистика, 1977. – 128 с.

Application of Conceptual Graphs in Digital Libraries

M. Bogatyrev, V. Latov, I. Stolbovskaya

Some results of application conceptual graphs as a content of digital libraries are presented. The problem of clustering of conceptual graphs and its decision are considered.

♣ Работа выполнена при поддержке гранта РФФИ № 07-07-00276-а.

.