ОНТОЛОГО-ОРИЕНТИРОВАННАЯ КЛАСТЕРИЗАЦИЯ ПОЛЬЗОВАТЕЛЕЙ НА ОСНОВЕ ИСТОРИИ ИХ РАБОТЫ С СИСТЕМОЙ УПРАВЛЕНИЯ ЗНАНИЯМИ

А.М. Кашевник

УДК 004.8

Кашевник А. М. **Онтолого-ориентированная кластеризация пользователей на основе истории их работы с системой управления знаниями.**

Аннотация. Управление знаниями — это совокупность процессов, которые управляют созданием, извлечением, обработкой, использованием, распространением, использованием знаний и предоставлением доступа к ним в некоторой предметной области. Система управления знаниями представляет собой комплекс процедур, реализующих эти процессы. Для представления знаний в настоящее время в России и в мире широко используются онтологии. Неотъемлемой частью любой системы взаимодействующей с пользователем является возможность персонифицировать поток информации и знаний между системой и пользователем. В работе предложен метод онтолого-ориентированной кластеризации для группировки пользователей системы управления знаниями на основе их предпочтений. Такая группировка позволяет выявлять общие предпочтения групп пользователей и адаптировать поток информации и знаний в зависимости от этих предпочтений.

Ключевые слова: кластеризация пользователей, профили пользователей, управление знаниями, онтологии.

Kashevnik A. M. Ontology**-oriented user clustering based on the history of interaction between user and knowledge management system**.

Abstract. Knowledge management is a combination of the processes which manage creation, extraction, processing, utilization, and distribution of knowledge in a certain domain. The knowledge management system is a complex of the procedures which realize these processes. Ontologies are widely used today for the knowledge representation both in Russia and abroad. A part of any system interacting with user is a possibility to personify the information and knowledge flow between the system and the user. A method of ontology-oriented clustering for grouping knowledge management system users based on their preferences is proposed. Such grouping makes it possible to reveal common preferences of user groups and to adapt the information and knowledge flow based on these preferences.

Keywords: user clustering, user profiles, knowledge management, ontologies

1. Введение. Управление знаниями представляет собой совокупность процессов, которые управляют созданием, извлечением, обработкой, использованием, распространением, использованием знаний и предоставлением доступа к ним в некоторой предметной области. Система управления знаниями представляет собой комплекс процедур, реализующих эти процессы. Для представления знаний в настоящее время в России и в мире широко используются онтологии. Онтология

— это подробная спецификация модели предметной области; она включает в себя словарь (т. е. список логических констант и предикатных символов) для описания предметной области и набор логических высказываний, формулирующих существующие в данной проблемной области ограничения и определяющих интерпретацию словаря.

Неотъемлемой частью систем, взаимодействующих с пользователями, является возможность персонифицировать поток информации и знаний между системой и пользователем. Для этих целей могут быть применены профили пользователей, которые включают в себя различную информацию, характеризующую пользователя, а также историю его работы с системой.

В работе предложен метод онтолого-ориентированной кластеризации для группировки пользователей системы управления знаниями на основе их предпочтений. Такая группировка позволяет выявлять общие предпочтения групп пользователей и адаптировать поток информации и знаний с этими пользователями в зависимости от этих предпочтений.

2. Формализм объектно-ориентированных сетей ограничений. В качестве средства формального описания знаний выбрана модель объектно-ориентированных сетей ограничений [1]. Знания представляются множествами классов, атрибутов классов, доменов атрибутов и ограничений, описанными средствами формализма объектно-ориентированных сетей ограничений.

В соответствии с выбранным формализмом онтология (O) описывается следующим образом:

$$O = \langle C, A, D, R \rangle,$$

где C — множество классов; A — множество атрибутов классов; D — множество доменов (областей допустимых значений) атрибутов; R — множество ограничений:

$$R = R^{I} \bigcup R^{II} \bigcup R^{III} \bigcup R^{IV} \bigcup R^{V} \bigcup R^{VI},$$

в которое входят следующие ограничения:

$R^{I} = \{r^{I}\}$, $cr^{I}a; c \in C, a \in A$ — описывают принадлежность атрибутов классам;

$R^{II} = \{r^{II}\}$, $(cr^{I}a)r^{II}d; c \in C, a \in A, d \in D$ — описывают принадлежность доменов атрибутам;

$R^{III} = \{r^{III}\}$, $c' r^{III} c''; c', c'', \in C$ — задают совместимость классов (структурные ограничения совместимости классов);

$R^{IV} = \{r^{IV}\},\ c' r^{IV} c'';c' \in O, c'' \in O, c' \neq c''$ — описывают иерархические связи между классами (иерархические структурные ограничения) и включают в себя два типа отношений: 1) «быть экземпляром» (определяют таксономию классов), 2) «быть частью» (определяют иерархию классов);

$R^{V} = \{r^{V}\},\quad c' r^{V} c'', c', c'' \in C$ — описывают ассоциативные связи между классами (структурные ограничения одного уровня);

$R^{VI} = \{r^{VI}\},\quad r^{VI} = f(\{c\},\{c,a\}) = \text{True} \vee \text{False},\quad |\{c\}| \geq 0,$ $|\{a\}| \geq 0,\ c \in C,\ a \in A$ — функциональные ограничения, которые описывают функциональные отношения между классами и атрибутами.

3. Модель профиля пользователя в системе управления знаниями. Анализ существующих различных моделей профилей пользователей показал, что большинство таких моделей включают в себя следующие данные: имя, фамилию, пол, дату рождения, языки, которыми владеет пользователь; контактную информацию для связи с пользователем (номер телефона, электронную почту, номер для обмена сообщениями, Интернет-сайт, а также должность пользователя).

Так как профиль пользователя разрабатывается для системы управления знаниями, то:

1. для определения задач, в решении которых пользователь компетентен в данный момент времени, предлагается использовать атрибут *«роль»*;
2. для обеспечения конфиденциальности некоторых знаний в системе предложен атрибут *«уровень доступа»*, определяющий ту информацию и знания, к которым данный пользователь имеет доступ;
3. для работы с группами пользователей предложен атрибут «группа», который определят принадлежность пользователя к той или иной группе на основе пользовательских предпочтений;
4. для определения потенциальных возможностей пользователя предложен атрибут «оборудование», который определяет окружение пользователя относительно аппаратного и программного обеспечения;
5. для возможности запретить просмотр профиля пользователя другими пользователями системы предложен атрибут «ви-

димость профиля», с помощью которого пользователь может сделать свой профиль невидимым для других;

6. системе управления знаниями может потребоваться информация о том, где находится пользователь в данный момент времени, для этой цели предложен атрибут «местоположение пользователя», содержащий в себе информацию о текущем географическом положении пользователя;
7. для определения, может ли в данный момент пользователь решать задачу, предложен атрибут «Часовой пояс», определяющий как скоро у пользователя начнется рабочий день.

Для оценки действий пользователя в рамках системы предложены атрибуты: «исполнительность», «старание» и «компетентность», которые заполняются другими пользователями, взаимодействующими с данным пользователем.

При функционировании системы управления знаниями важно использовать предпочтения пользователя. Для этих целей ~~были~~ предложены атрибуты: «Классы», соответствующие классам онтологии; «атрибуты классов», соответствующие атрибутам онтологии; и «значения атрибутов», соответствующие значениям атрибутов онтологии.

Для последующего анализа информации о пользователе (в том числе и методом онтолого-ориентированной кластеризации, см. раздел 3) предлагается хранить в профиле пользователя все его запросы, сформированные на основе их контексты, а также информацию, характеризующую пользователя на момент инициализации запроса (контексты пользователя). Информационная модель профиля пользователя системы управления знаниями представлена на **Ошибка! Источник ссылки не найден.** и содержит следующие элементы [2]:

1. «Контекст пользователя» — информация о пользователе 1) персональная 2) системная информацию, 3) контактная (обратная связь) и 4) предпочтения пользователя.
2. «История запросов» — все запросы пользователя, связанные с ними сформированные системой контексты (для обработки этих запросов [3]), а также контекст пользователя на момент инициализации его запроса.

Рис. 1. Информационная модель профиля пользователя.

4. Метод онтолого-ориентированной кластеризации. Представленный ниже метод онтолого-ориентированной кластеризации позволяет:

1) группировать пользователей системы управления знаниями на основе их предпочтений;
2) выявлять предпочтения пользователей системы управления знаний.

Входными данными для метода онтолого-ориентированной кластеризации являются запросы пользователей и контексты, представляющие собой срезы онтологии, соответствующие этим запросам. Данная информация содержится в профилях пользователей.

Общая схема метода группировки пользователей представлена на рис. 1. На первом шаге по запросам пользователей на основе контекстов, соответствующих этим запросам, строится математическая модель, связывающая эти элементы. В качестве модели предложено использовать взвешенный граф. На втором шаге, выполняются преобразования этой модели, позволяющие установить связи между каждой парой запросов. На третьем шаге производится содержательная интерпретация полученного преобразования на предметную область, т. е. строятся группы пользователей.

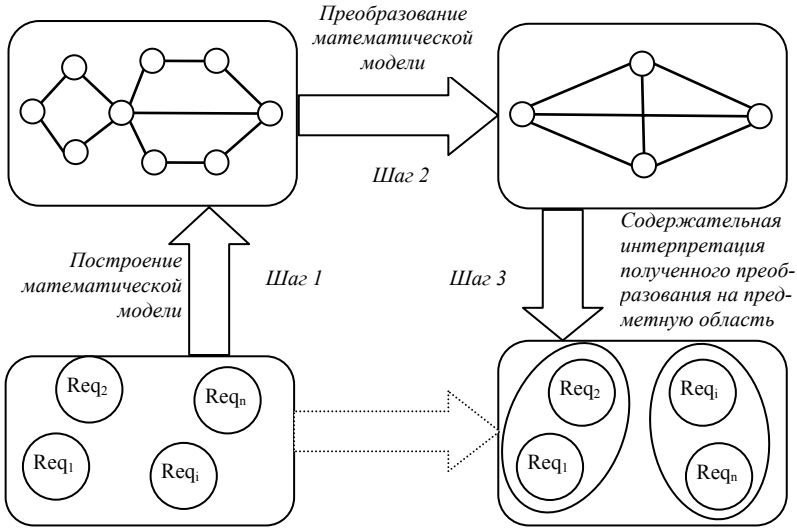

Рис. 1. Общая схема метода группировки пользователей.

На первом шаге (рис. 2) производится операция композиции над запросами пользователей и элементами соответствующим им контекстам. Строится взвешенный граф G_0 (рис. 3):

$G_0 = \langle N, E\rangle = \langle (c, a, Req), (ca, cc, cReq, aReq)\rangle$,

где N — вершины графа, а E — дуги.

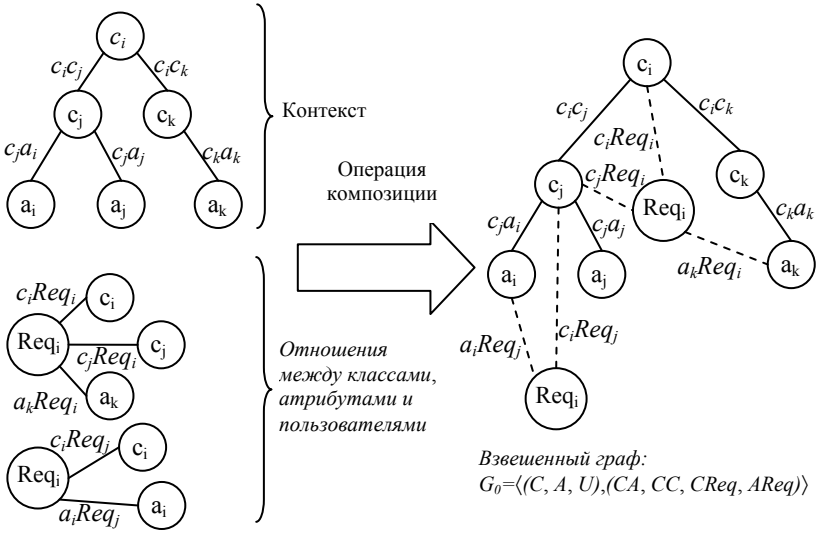

Рис. 2. Построение математической модели на основе запросов участника и соответствующим им контекстам.

Граф имеет три типа вершин: c — класс, a — атрибут и Req — запрос. Дуги графа G_0 могут быть представлены следующей матрицей:

$$T[i,j] = \begin{cases} 0, \text{ if } i = j, \\ c_{ij} \in \Re^{+}, \text{если существует дуга из } i \text{ в } j, \\ \infty, \text{ если не существует дуги из } i \text{ в } j; \end{cases}$$

Можно выделить два типа дуг на графе G_0. Тип I (ca, cc) определяется иерархией классов и атрибутов онтологии. Тип II ($cReq$, $aReq$) определяется отношениями между запросом и классом (атрибутом).

Веса дуг между вершинами, представляющими классы и запросы $cReq_{weight}$ и атрибуты и запросы $aReq_{weight}$, определяются через схожесть терминов запроса и терминов класса (атрибута):

$$c\,\mathrm{Re}\,q_{weight} = 1 - c\,\mathrm{Re}\,q_{sim}$$
$$a\,\mathrm{Re}\,q_{weight} = 1 - a\,\mathrm{Re}\,q_{sim},$$

где $cReq_{sim}$ — похожесть запроса термину класса c, $aReq_{weight}$ — схожесть запроса термину атрибута a.

Все дуги ca и cc, соединяющие классы и атрибуты, имеют веса: ca_{weight}, $cc_{weight} \in [\varepsilon, 1)$, которые определяются инженером по онтологии. Параметр cc_{weight} показывает вес дуги между классами в онтологии, а ca_{weight} — вес дуги между классом и атрибутом (рис. 3).

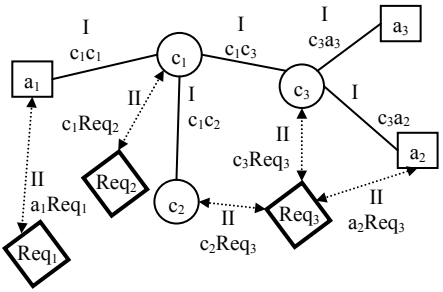

Рис. 3. Взвешенный граф для процедуры кластеризации запросов пользователя.

На втором шаге (рис. 4) производится преобразование математической модели для нахождения кратчайших расстояний между каждой парой запросов пользователя. Для этих целей ~~был~~ адаптирован алгоритм Флойда [4], который используется для нахождения кратчайших расстояний между каждой парой вершин на графе.

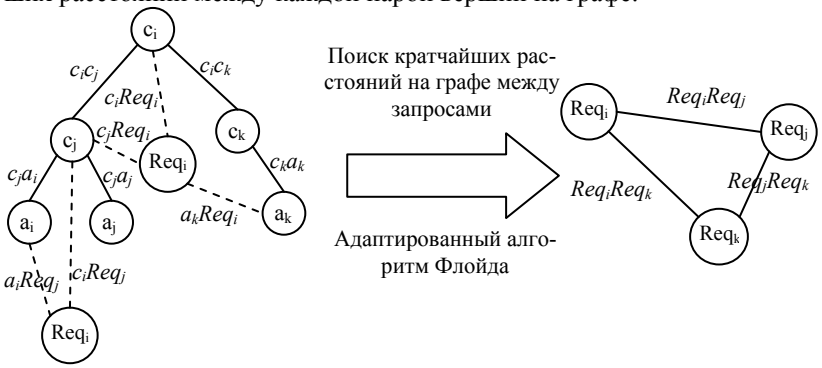

Рис. 4. Преобразование математической модели.

В рассматриваемом случае достаточно знать только веса кратчайших путей между каждой парой запросов пользователя, по этой причине алгоритм был несколько упрощен. Построим граф G_1, состоящий из запросов и взвешенных дуг между ними (рис. 5), изменив алгоритм Флойда.

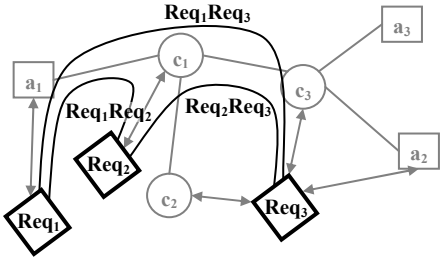

Рис. 5. Граф G_1, состоящий из запросов и взвешенных дуг между ними.

Для алгоритма будут использоваться следующие данные: матрица T, описывающая дуги графа G_0, и константа p, равная количеству дуг в графе G_0.

Алгоритм, представленный ниже в псевдокоде, рассчитывает веса кратчайших путей между запросами в графе G_0:

```
for i from 1 to p do
 for j from 1 to p do
  for k from 1 to p do
   if i<>j and T[i,j]<> ∞ and i<>k and T[i,k]<> ∞ and
     (T[j,k]= ∞ or T[j,k]>T[j,i]+T[i,k]) then
       T[j,k] = T[j,i]+T[i,k]
   end if
  end for
 end for
end for
```

В результате работы алгоритма матрица T содержит веса кратчайших путей между каждой парой запросов. На основе этой матрицы можно построить граф G_1 (см. рис. 5).

На третьем шаге (рис. 6) производится содержательная интерпретация полученного преобразования на предметную область с помощью кластеризации.

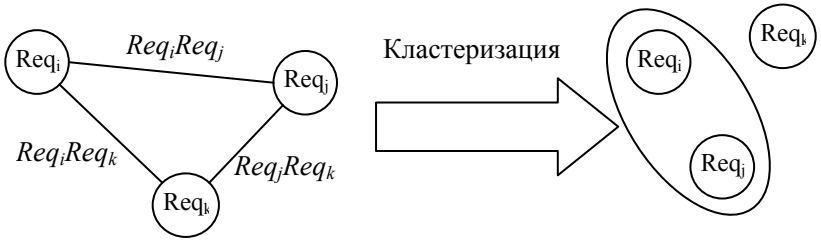

Рис. 6. Содержательная интерпретация полученного преобразования.

Для того чтобы разбить запросы пользователей на кластеры, достаточно разделить граф G_1 на подграфы G_1^i, $i = 1, \ldots, n$, где n — количество кластеров.

Определим массу кластера $Dist[G_1^i]$ как сумму весов всех дуг подграфа G_1^i. Оптимальная кластеризация может быть достигнута при удовлетворении следующих условий:

1. $n \rightarrow min$, т. е. необходимо минимизировать число кластеров,
2. $W_{max} > Dist[G_1^i]$, $i = 1, \ldots, n$, т. е. максимальная масса кластера для каждого подграфа меньше чем константа W_{max}, (определяемая администратором системы управления знаниями, на основе практического использования системы).

Алгоритм выглядит следующим образом:

1. $Dist[Req_i] = 0$, $i = 1, \ldots, n$. На начальном этапе алгоритма каждая вершина считается подграфом. Масса каждого такого подграфа $Dist[Req_i] = Dist[G_1^i]$ равна нулю.
2. Заполняется вектор V, таким образом, что: $V[z] = ARC_{weight} + Dist[G_1^i] + Dist[G_1^j]$; т. е., каждый элемент $V[z]$ вектора V равен сумме следующих весов: веса дуги между Req_i и Req_j (ARC_{weight}) и массы этих подграфов ($Dist[G_1^i]$ и $Dist[G_1^j]$).
3. Выбирается минимальный элемент $V[z]$ из вектора V.
4. Если $V[z] > W_{max}$, то алгоритм завершается, а текущее разбиение и есть искомая кластеризация, удовлетворяющая заданным условиям, так как дальнейшее объединение ~~далее~~ подграфов невозможно по причине того, что минимальная масса полученного при объединении подграфа будет больше, чем максимальная масса кластера (W_{max}). Если $V[z] < W_{max}$, поэтому выполняется шаг пять.

5. Подграфы G_1^i и G_1^j, соответствующие элементу $V[z]$, объединяются в G_1^i, и масса нового подграфа $Dist[G_1^i]$ будет равна $V[z]$. Подграф G_1^j и вектор $V[z]$ удаляются.
6. Обновляются значения вектора V для дуг смежных с подграфом G_1^i (если подграф G_1^k смежен с подграфом G_1^i, то элемент $V[z]$, соответствующий дуге между подграфами G_1^i и G_1^k будет равен $A[i, k] = ARC_{weight}[i, k] + Dist[G_1^i] + Dist[G_1^k]$).
7. Переход на шаг три.

После окончания работы алгоритма будут сформированы группы запросов (подграфы) (рис. 7). Так как каждый запрос однозначно определяет пользователя, следовательно, и получаются группы пользователей.

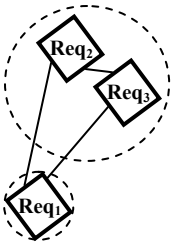

Рис. 7. Группы запросов пользователей.

5. Оценка сложности метода онтолого-ориентированной кластеризации. Сложность (P) метода группирования пользователей на основе их предпочтений может быть определена как сумма сложностей алгоритмов, составляющих метод (построение графа «класс—пользователь» — P_1, преобразование графа — P_2, алгоритм кластеризации графа — P_3), поскольку они выполняются последовательно.

Рассмотрим сложность каждого из них. Обозначим: N — количество вершин графа «класс—пользователь»; n — количество вершин графа пользователей; L — максимальное количество классов, встречающихся в одном запросе.

Сложность построения графа «класс—пользователь» определяется следующим образом. Алгоритм извлекает все запросы пользователей за требуемый период. Далее для каждого запроса извлекаются все классы и атрибуты, которым он соответствует, и пересчитываются веса. Алгоритм использует два вложенных цикла: 1) внешний по N, 2) внутренний по L. Следовательно, сложность построения графа определяется по формуле

$$P_1 = O(NL),$$

где символ O для оперирования с приближенными величинами определен в работе [5].

Усеченный алгоритм Флойда, используемый в задаче, находит веса кратчайших путей между всеми вершинами графа. Алгоритм использует три вложенных цикла по N, следовательно, его сложность определяется по формуле

$$P_2 = O(N^3)$$

Рассмотрим отдельно сложность шагов алгоритма, входящих в цикл и не входящих в него.

Первый шаг, не входящий в цикл, даёт сложность $O(1)$. Далее мы строим вектор V (второй шаг алгоритма) - сложность $O(n^2)$ максимальная сложность (в случае полносвязного графа).

Рассмотрим сложность шагов алгоритма, входящих в цикл. Поиск минимума в векторе V, размерностью n (третий шаг алгоритма) — $O(n)$ операций. Проверка на четвертом шаге алгоритма $O(1)$. Пятый шаг — $O(1)$. На шестом шаге обновление вектора V для дуг, смежных с некоторой вершиной. В случае, когда граф связный, сложность этого шага будет наибольшей, равной $O(n)$. Шаги 3-6 выполняются в цикле до тех пор, пока не будет достигнуто оптимальное разбиение. В предельном случае, алгоритм будет выполняться до тех пор, пока не получится один кластер, состоящий из всех пользователей, сложность будет наибольшая — $O(n^2)$, во всех остальных случаях сложность будет меньше.

В результате получается, что сложность алгоритма кластеризации определяется как сумма сложностей:

$$P_3 = O(n^2) + O(n^2),$$

следовательно:

$$P_3 = O(n^2).$$

Тогда сложность сценария кластеризации будет следующая:

$$P = P_1 + P_2 + P_3 = O(N \cdot L) + O(N^3) + O(n^2).$$

Учитывая, что $N \geq n$, то третьей частью можно пренебречь. Тогда

$$P = O(NL) + O(N^3).$$

При больших количествах запросов количество классов, встречающихся в одном запросе, существенно меньше, чем количество этих запросов ($L << N$). Следовательно, можно заменить L на N в вышеприведенной формуле Тогда $P = O(N^2) + O(N^3)$.

Следовательно, $P = O(N^3)$.

То есть сложность сценария кластеризации порядка N^3 определяется для систем с большим количеством запросов пользователей.

6. Заключение. Представленный в статье метод онтолого-ориентированной кластеризации позволяет группировать пользователей системы управления знаниями на основе их предпочтений. Входными данными для метода онтолого-ориентированной кластеризации являются запросы пользователей и контексты, представляющие собой срезы онтологии, соответствующие этим запросам. Эта информация содержится в профилях пользователей. Сложность метода составляет $O(N^3)$, N — количество запросов пользователей, при большом N.

Литература

1. *Смирнов А.В.*, *Левашова Т.В. Пашкин М.П.*, *Шилов Н.Г.* Онтолого–ориентированный многоагентный подход к построению систем интеграции знаний из распределенных источников // Информационные технологии и вычислительные системы. 2002. № 1. С. 62–82.
2. *Кашевник А.М.* Профилирование в системах управления корпоративными знаниями // Материалы IX Санкт-Петербургской междунар. конф. «Региональная информатика 2004», 22–24 июня 2004, СПб.
3. *Смирнов А.В.*, *Шилов Н.Г.*, *Кашевник А.М.* Персонифицированная контекстно-ориентированная поддержка взаимодействия участников производственных сетей // Искусственный интеллект. 2009. Т. 1. С. 46–56.
4. *Floyd R.W.* Algorithm 97 — Shortest path // Comm. of ACM. 1962. 5. 345 с.
5. *Кнут Д.* Искусство программирования для ЭВМ. М.: Мир, 1976.

Кашевник Алексей Михайлович — канд. техн. наук, старший научный сотрудник лаборатории интегрированный систем автоматизации учреждения Российской академии наук Санкт-Петербургского института информатики и автоматизации РАН (СПИИРАН). Область научных интересов: управление знаниями, профилирование, онтологии, интеллектуальный пространства. Число научных публикаций — 45. alexey@iias.spb.su; СПИИРАН, 14-я линия, д.39, Санкт-Петербург, 199178, РФ; р.т. +7(812)328-8071, факс +7(812)328-0685.

Kashevnik Alexey — Ph.D.; senior researcher of the laboratory of computer aided integrated systems institution of the Russian Academy of Sciences St.Petersburg Institute for Informatics and Automation of RAS (SPIIRAS). Research area: knowledge management, profiling, ontologies, smart-spaces. Number of publications — 45. alexey@iias.spb.su; SPIIRAS, 14th Line V.O., 39, Saint-Petersburg, 199178, Russia; office phone +7(812)328-8071, fax +7(812)328-0685.

Рекомендовано лабораторией ИСА, зав. лаб. А.В. Смирнов, д-р техн. наук, проф.
Статья поступила в редакцию 02.12.2009.

РЕФЕРАТ

Кашевник А. М. **Онтолого-ориентированная кластеризация пользователей на основе истории их работы с системой управления знаниями.**

Управление знаниями — это совокупность процессов, которые управляют созданием, извлечением, обработкой, использованием, распространением, использованием знаний и предоставлением доступа к ним в некоторой предметной области. Система управления знаниями представляет собой комплекс процедур, реализующих эти процессы. Для представления знаний в настоящее время в России и в мире широко используются онтологии. Неотъемлемой частью любой системы взаимодействующей с пользователем является возможность персонифицировать поток информации и знаний между системой и пользователем.

Данная работа посвящена разработке метода онтолого-ориентированной кластеризации пользователей на основе истории их работы в рамках системы управления знаниями. В работе используются методы искусственного интеллекта для работы со знаниями, технологии профилирования пользователей и кластеризации.

Предложенный в работе метод онтолого-ориентированной кластеризации для группировки пользователей системы управления знаниями позволяет выявлять общие предпочтения групп пользователей и адаптировать поток информации и знаний в зависимости от этих предпочтений. Входными данными для метода являются запросы пользователей и контексты, представляющие собой срезы онтологии, соответствующие этим запросам. Эта информация содержится в профилях пользователей. Сложность метода определяется как $O(N^3)$, где N — количество запросов пользователей, при большом N.

Данный метод может быть широко применен в ориентированных на пользователя системах, основанных на знаниях. Это могут быть экспертные системы, интеллектуальные поисковые системы, системы управления производственными сетями и т. п. Персонификация таких систем в современном мире очень важна, так как позволяет автоматизировать процессы взаимодействия системы с пользователем, пользователя с пользователем, протекающие в таких системах.

SUMMARY

Kashevnik A. M. Ontology**-oriented user clustering based on the history of interaction between user and knowledge management system**.

Knowledge management is a combination of the processes which manage creation, extraction, processing, utilization, and distribution of knowledge in a certain domain. The knowledge management system is a complex of the procedures which realize these processes. Ontologies are widely used today for the knowledge representation both in Russia and abroad. A part of any system interacting with user is a possibility to personify the information and knowledge flow between the system and the user.

In this paper the method of ontology-oriented clustering for grouping knowledge management system users based on their preferences is proposed. The knowledge-based methods of artificial intelligence, profiling and clustering are used in this paper.

Described in the paper method of ontology-oriented user clustering allows to identify groups of users and common preferences of these groups. It allows adapting information and knowledge flow between user and system based on these preferences. Input data for this method is a user requests and ontology slices which corresponded to these requests. This information is keeping in the user profile. The complexity of this method is $O(N^3)$, N — count of user requests, for the large amount of user requests.

This method can be widely applied in the user-oriented knowledge-based systems. For example it can be applied in expert systems, intellectual searching systems, knowledge management systems for production networks and other. Personification of such systems is very important task in modern life, because it allows automating the processes of interaction system with user and interaction between users in these systems.