УДК: 519.6

ОЦЕНКА РЕЛЕВАНТНОСТИ ДОКУМЕНТОВ ОНТОЛОГИЧЕСКОЙ БАЗЫ ЗНАНИЙ

БОСКЕБЕЕВ К.ДЖ., ХИЖНЯК М.А.
izvestiya@ktu.aknet.kg

В статье предложено несколько мер релевантности ролевых кластеров документа, формализующих близость семантических сетей поискового образа документа и семантических сетей запроса. На основе указанных мер предложен алгоритм оценки релевантности документа запросу.

Цель исследования. В общей постановке о задаче поиска информации следует говорить в терминах модели поиска, которая включает в себя способ представления документов, способ представления поисковых запросов, вид критерия релевантности документов.
Модель исследования. В статье существенно используется «важность» концептов в семантической сети рассматриваемой онтологической базы знаний. Важной составной частью предлагаемой методики оценки релевантности документа является построение семантической сети этого документа.

Представим семантическую сеть $S(O)$ рассматриваемой онтологии O в виде взвешенного связного мультиграфа $G(O)$. Узлы этого графа соответствуют концептам множества $C(O) = \{c_i, i \in [1:n^O]\}$, а ребра – четким бинарным отношениям между ними, каждое из которых принадлежит одному из типов $U_p, p \in [1:m^O]$.

Определены веса $w_i^O, \ i \in [1:n^O]$ узлов графа $G(O)$, формализующие «важность» соответствующих концептов в сети $S(O)$. Для каждого из ребер $(c_i, c_j), \ i, j \in [1:n^O], \ i \neq j$ графа $G(O)$ полагается заданным также $(1 \times m^O)$ – вектор весов $\{v_{i,j,p}^O, p \in [1:m^O]\}$, где $v_{i,j,p}^O = 0$, если концепты (c_i ,c_j) не связаны между собой отношением типа U_p, и $v_{i,j,p}^O = v_p^O$ – в противном случае. Здесь v_p^O - априори заданный вес отношений типа U_p в онтологии O.

Некоторые методы определения весов концептов w_i^O и весов отношений v_p^O предложены в работе [3]. Для определения весов концептов может быть также использована их семантическая близость, полученная с помощью соответствующего словаря [5]. Веса концептов могут быть сформированы также на основе понятий центральности по близости и центральности по посредничеству [6].

Перейдем, например, с помощью аддитивной свертки

$$v_{i,j}^O = \sum_p \lambda_p v_{i,j,p}^O, \ p \in [1:m^O] \qquad (1)$$

от взвешенного мультиграфа $G(O)$ к взвешенному обыкновенному графу, в котором вес ребра (c_i ,c_j) равен $v_{i,j}^O$. Сохраним за полученным графом прежнее обозначение. Здесь и далее $\lambda_p, \ p = 1,2,...$ – положительный скалярный вещественный множитель, определяющий относительный вес компонентов аддитивной скалярной свертки вида (1).

Аналогично определим семантическую сеть $S(T) \subset S(O)$ документа T в виде связного взвешенного обыкновенного графа $G(T)$. Узлы этого графа соответствуют $n^T \leq n^O$ концептам

$C(T) \subset C(O)$ документа T, а ребра – связям между ними. Вес узла графа $G(T)$, соответствующего концепту $c_i \in C(T)$, обозначим w_i^T, а атрибуты его ребра (c_i ,c_j) зададим парой $(l_{i,j}^T; v_{i,j}^T)$, где $l_{i,j}^T$ – «расстояние» между узлами c_i ,c_j , а $v_{i,j}^T$ - вес ребра (c_i ,c_j) .

В терминах графа $G(T)$ задача построения семантической сети документа $S(T)$ сводится к решению двух следующих задач.

Задача 1 (определения топологии графа $G(T)$). По каким правилам связывать узлы этого графа ребрами, т.е. устанавливать связи между концептами множества $C(T)$?

Задача 2 (определения весов узлов и атрибуты ребер графа $G(T)$). Исходя из каких соображений, назначать веса w_i^T узлов этого графа, а также атрибуты $l_{i,j}^T$, $v_{i,j}^T$ его ребер?

Определение топологии графа $G(T)$. В общей постанове эту задачу следует отнести к задаче огрубления графа.

Классические методы решения задачи огрубления графа основаны на итерационном стягивании смежных узлов графа G^{α} в узлы графа $G^{\alpha+1}$, где $\alpha = 0,1,2,\ldots$ - номер итерации, $G^0 = G(O)$. В результате этого процесса ребро между двумя вершинами графа G^{α} удаляется и создается мультиузел графа $G^{\alpha+1}$, объединяющий оба стягиваемых узла. Задача огрубления графа $G(O)$ до графа $G(T)$ имеет ту специфику, что ни на одной из итераций указанного итерационного процесса в один узел не могут быть стянуты те узлы графа G^{α}, которые принадлежат графу $G(T)$.

Обычно задача огрубления графа решается в терминах паросочетаний. *Паросочетанием* в графе называется набор его ребер, в котором любые два ребра не инцидентны общему узлу. Таким образом, граф $G^{\alpha+1}$строится на основе графа G^{α} путем нахождения в графе G^{α} паросочетания и стягивания в мультиузел узлов, входящих в каждую из пар этого паросочетания. Непарные узлы графа G^{α} просто копируются в граф $G^{\alpha+1}$. Важно, что граф, огрубленный, с использованием паросочетаний, сохраняет многие свойства исходного графа. Так, например, если граф G^0 является планарным, то граф G^{α} также планарен.

В терминах паросочетаний специфика нашего случая состоит в том, что любая пара узлов каждого из паросочетаний не может включать в себя одновременно два узла графа $G(T)$.

С точки зрения повышения эффективности процесса огрубления графа целесообразно использовать *насыщенные паросочетания* – паросочетания, в которых хотя бы один узел любого ребра, не вошедшего в паросочетание, инцидентен ребру, вошедшему в паросочетание. Вообще говоря, с той же точки зрения желательным является использование *максимальных паросочетаний* - насыщенных паросочетаний, которые имеют максимальное число ребер. Однако вычислительная сложность формирования максимальных паросочетаний в общем случае значительно выше аналогичной вычислительной сложности для просто насыщенных паросочетаний. Поэтому обычно в вычислительной практике ограничиваются последними [1].

Утверждение 1. Оценка снизу количества итераций, необходимых для построения графа $G(T)$, с использованием насыщенных паросочетаний равна $\log\left(n^O/n^T\right)$.

Справедливость утверждения следует из того факта, что при использовании насыщенных паросочетаний число узлов графа $G^{\alpha+1}$не может быть, очевидно, меньше половины числа узлов графа G^{α}.

Наиболее известны три следующих метода построения насыщенных паросочетаний: случайное паросочетание (*RM*); паросочетание из тяжелых ребер (*HEM*); паросочетание из тяжелых клик (*HCM*) [5].

Случайное паросочетание на итерации α строится по следующей схеме:

1) все узлы C^{α} текущего графа G^{α} объявляем немаркированными;

2) случайным образом выбираем немаркированный узел, еще не включенный в паросочетание, − пусть это будет узел c_i^{α} ;

3) из числа немаркированных узлов, смежных узлу c_i^α, случайным образом выбираем узел (пусть это будет узел c_j^α), также еще не включенный в паросочетание;

4) если оба узла или один из узлов пары c_i^α, c_j^α не принадлежат графу $G(T)$, то включаем ребро (c_i^α, c_j^α) в паросочетание, и узлы c_i^α, c_j^α маркируем;

5) если ни одного немаркированного узла, смежного узлу c_i^α, не существует, то узел c_i^α маркируем и оставляем свободным (чтобы затем перенести его в граф $G^{\alpha+1}$);

6) если в графе G^α имеются еще немаркированные узлы, то переходим к шагу 2.

Данную схему иллюстрирует рисунок 1, на котором слева показан граф $G^{\alpha-1}$ и сформированное на его основе паросочетание, а справа – граф G^α.

*Паросочетание из тяжелых ребер***.** Схема построения этого паросочетания отличается от рассмотренной выше схемы шагом 3, который в данном случае формулируется следующим образом. Из числа немаркированных узлов, смежных узлу c_i^α, выбираем такой узел c_j^α, еще не включенный в паросочетание, что вес ребра (c_i^α, c_j^α) является максимальным среди весов всех возможных ребер, связанных с узлом c_i^α.

Паросочетание из тяжелых клик. В данном случае также меняется только шаг 3 рассмотренной схемы формирования случайного паросочетания: из числа немаркированных узлов, смежных узлу c_i^α, выбираем такой узел c_j^α, еще не включенный в паросочетание, что реберная плотность мультиузла, который получается стягиванием узлов c_i^α, c_j^α, является максимально возможной по сравнению со всеми иными вариантами выбора узла c_j^α.

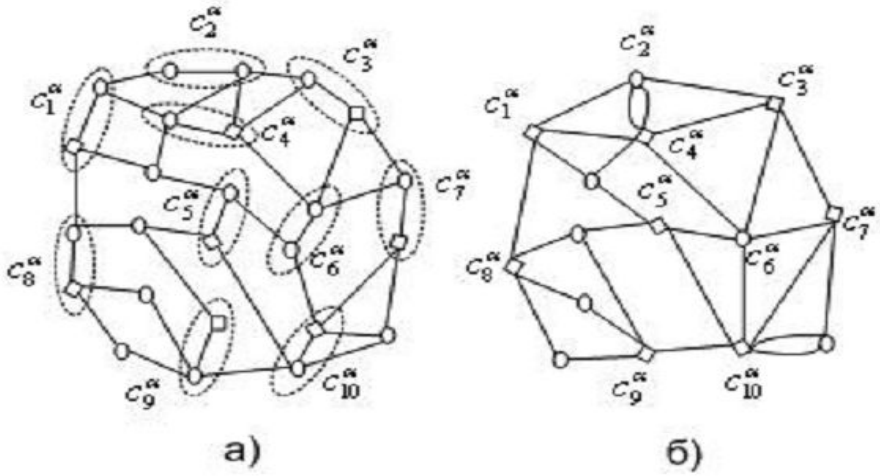

Рисунок 1. К методу случайных паросочетаний: квадратиками показаны узлы графа G(T) : а) граф $G^{\alpha-1}$; б) граф G^α.

Итерации во всех рассмотренных методах формирования паросочетания заканчиваются, когда в результате данной итерации не удалось выделить ни одной пары узлов. Другими словами, итерации заканчиваются, если в текущем графе G^α содержатся только узлы графа G(T).

Отметим следующее обстоятельство. В силу наличия элемента случайности при формировании паросочетаний различные итерационные процессы порождают, вообще говоря, графы G(T), имеющие различную топологию. Таким образом, возникает задача

получения в некотором смысле наилучшего графа G(T). При этом в качестве максимизируемого критерия оптимальности графа можно использовать, например, его реберную плотность (коэффициент кластеризации) [2].

Определение весов узлов и ребер графа G(T). Выделим два случая:

1) рассматриваемая пара узлов паросочетания включает в себя узел, принадлежащий графу G(T) (например, пара c_1^α на рисунке 1);

2) пара узлов содержит только узлы, не принадлежащие графу G(T) (например, пара c_2^α на том же рисунке).

Случай 1. Пусть рассматриваемая пара включает в себя узел (или мультиузел) $c_i^\alpha \in C(T)$ и узел (или мультиузел) $c_j^\alpha \notin C(T)$, веса которых равны w_i^α, w_j^α соответственно, а атрибуты ребра (c_i^α, c_j^α) определяется парой $(l_{i,j}^\alpha; v_{i,j}^\alpha)$. Отметим, что во веденных обозначениях надиндекс α указывает на то, что в процессе огрубления графа G(O) веса его узлов и ребер, вообще говоря, изменяются. Здесь и далее в данном разделе для простоты записи индекс Т в обозначениях опущен.

Будем полагать, что в процессе стягивания узлов c_i^α, c_j^α узел c_j^α стягивается к c_i^α, так что в результате получается мультиузел $c_i^{\alpha+1} \in C(T)$, вес которого равен $w_i^{\alpha+1}$. Условно результат данной процедуры будем записывать в виде $c_i^{\alpha+1} = c_i^\alpha \oplus c_j^\alpha$.

Логично исходить из того, что вес $w_i^{\alpha+1} = w_i^{\alpha+1}(w_i^\alpha, w_j^\alpha, l_{i,j}^\alpha, v_{i,j}^\alpha)$ является некоторой положительной возрастающей функцией своих аргументов w_i^α, w_j^α, $v_{i,j}^\alpha$ и такой же убывающей функцией аргумента $l_{i,j}^\alpha$. В простейшем случае в качестве такой функции может быть использована функция вида:

$$w_i^{\alpha+1}(w_i^\alpha, w_j^\alpha, l_{i,j}^\alpha, v_{i,j}^\alpha) = \lambda_1 w_i^\alpha + \lambda_2 w_j^\alpha \frac{v_{i,j}^\alpha}{l_{i,j}^\alpha}. \quad (2)$$

В результате стягивания узла c_j^α к узлу c_i^α атрибуты ребер, инцидентных узлу c_i^α, не меняются, а значения атрибутов ребер, инцидентных узлу c_j^α, должны быть по некоторому правилу изменены. Рассмотрим одно из таких ребер (c_j^α, c_p^α), атрибуты которого равны $(l_{j,p}^\alpha; v_{i,p}^\alpha)$. Это ребро заменяется на ребро $(c_i^{\alpha+1} = c_i^\alpha \oplus c_j^\beta, c_p^{\alpha+1} = c_p^\alpha)$, которому соответствуют атрибуты $(l_{i,p}^{\alpha+1}; v_{i,p}^{\alpha+1})$.

Естественно положить, что длина ребра $(c_i^{\alpha+1}, c_p^{\alpha+1})$ равна:

$$l_{i,p}^{\alpha+1} = l_{i,j}^\alpha + l_{j,p}^\alpha,$$

т.е. длина ребра (c_i^α, c_j^α) превышает длину ребра (c_j^α, c_p^α). Логично также принять, что вес ребра $v_{i,p}^{\alpha+1} = v_{i,p}^{\alpha+1}(v_{i,j}^\alpha, v_{j,p}^\alpha)$ является некоторой положительной возрастающей функцией своих аргументов. В простейшем случае можно положить:

$$v_{i,p}^{\alpha+1} = \lambda_1 v_{i,j}^\alpha + \lambda_2 v_{j,p}^\alpha. \quad (3)$$

Схему рассмотренного алгоритма иллюстрирует рисунок 2. Здесь принято, что:

$$w_i^{\alpha+1}(w_i^\alpha, w_j^\alpha, l_{i,j}^\alpha, v_{i,j}^\alpha) = w_i^\alpha + w_j^\alpha \frac{v_{i,j}^\alpha}{l_{i,j}^\alpha}, \quad (4)$$

$$v_{i,p}^{\alpha+1} = v_{i,j}^\alpha + v_{j,p}^\alpha. \quad (5)$$

Случай 2. Положим, что оба узла рассматриваемой пары (c_i^α, c_j^α) не принадлежат графу G(T), т.е. когда имеет место ситуация $c_i^\alpha, c_j^\alpha \notin C(T)$. Как и ранее, положим, что веса указанных узлов равны w_i^α, w_j^α, а атрибуты ребра (c_i^α, c_j^α) определяются парой $(l_{i,j}^\alpha; v_{i,j}^\alpha)$.

В этом случае также можно считать, что один из узлов (пусть это будет узел c_j^α) стягивается к другому узлу (c_i^α), так что в результате получается мультиузел $c_i^{\alpha+1} = c_i^\alpha \oplus c_j^\alpha$, $c_i^{\alpha+1} \notin C(T)$. Как и в предыдущем случае, положим, что вес этого узла равен $w_i^{\alpha+1}$ и представляет собой, например, функцию вида (2).

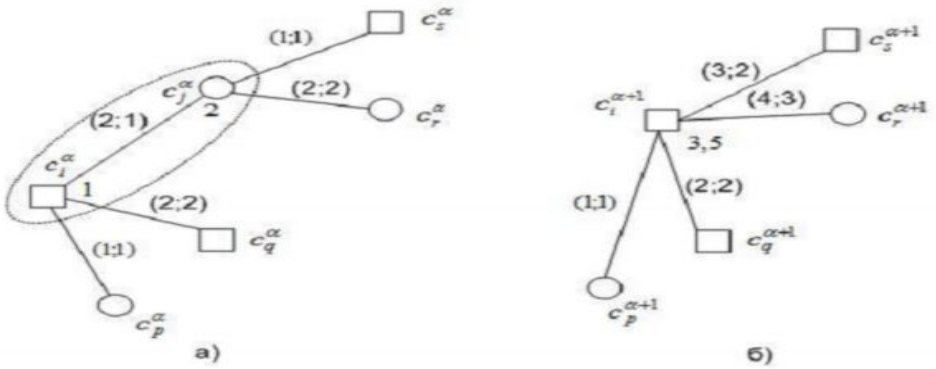

Рисунок 2. К стягиванию узла (мультиузла) $c_i^\alpha \in C(T)$ и узла (мультиузла) $c_j^\alpha \notin C(T)$: квадратиками показаны узлы графа G(T).

В отличие от предыдущего случая, здесь логично положить, что в результате стягивания узла c_j^α к узлу c_i^α меняются значения атрибутов всех ребер, инцидентных как узлу c_i^α, так и узлу c_j^α. Рассмотрим ребра (c_i^α, c_p^α), (c_j^α, c_q^α), атрибуты которых равны $(l_{i,p}^\alpha; v_{i,p}^\alpha)$, $(l_{j,q}^\alpha; v_{j,q}^\alpha)$ соответственно. Эти ребра заменяются на ребра

$(c_i^{\alpha+1}, c_p^{\alpha+1})$, $(c_i^{\alpha+1}, c_q^{\alpha+1})$, которым соответствуют атрибуты $(l_{i,p}^{\alpha+1}; v_{i,p}^{\alpha+1})$, $(l_{i,q}^{\alpha+1}; v_{i,q}^{\alpha+1})$, где:

$$l_{i,p}^{\alpha+1} = 0{,}5 l_{i,j}^{\alpha} + l_{i,p}^{\alpha}, \qquad l_{i,q}^{\alpha+1} = 0{,}5 l_{i,j}^{\alpha} + l_{j,q}^{\alpha}, \quad (6)$$

а веса $v_{i,p}^{\alpha+1}$, $v_{i,q}^{\alpha+1}$ определяются, например, по формуле вида (2).

Отметим, что принятые соглашения могут приводить к нецелым значениям расстояний между узлами графа $G^{\alpha+1}$, даже если все расстояния между узлами графа G^{α} являются целыми.

Схему рассмотренного алгоритма иллюстрирует рисунок 3. Здесь принято, что вес узла $c_i^{\alpha+1}$ определяется по формуле (4), веса ребер графа $G^{\alpha+1}$ – по формуле (5), а расстояние между узлами этого графа – по формуле (6).

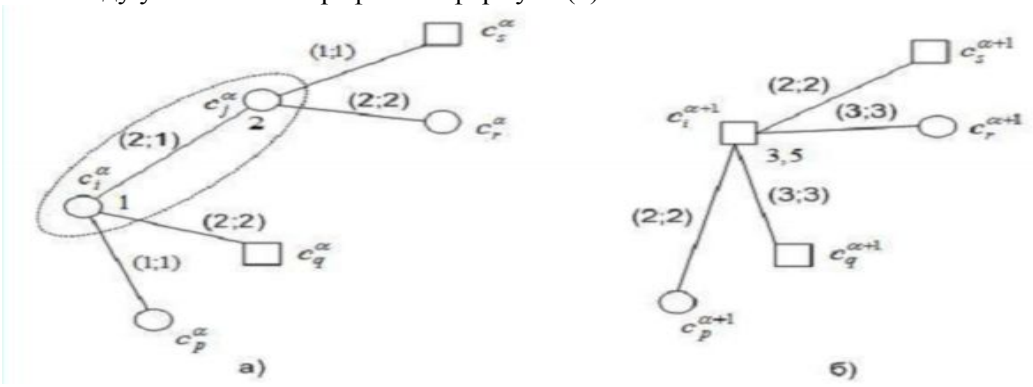

Рисунок 3. К стягиванию узлов (мультиузлов) $c_i^{\alpha}, c_j^{\alpha} \notin C(T)$.

В результате итерации α в графе $G^{\alpha+1}$ могут появиться кратные ребра (см., например, узлы c_2^{α}, c_4^{α} на рисунке 1а). Прежде чем переходить к основному циклу итерации) (α + 1), эти ребра следует объединить. Возникает вопрос, как вычислить значения атрибутов полученного ребра?

Положим, что двумя ребрами связаны узлы c_i^{α}, c_j^{α}, и атрибуты этих ребер равны $(l_{1,i,j}^{\alpha}, v_{1,i,j}^{\alpha})$, $(l_{2,i,j}^{\alpha}, v_{2,i,j}^{\alpha})$. В качестве расстояния между этими узлами $l_{i,j}^{\alpha+1}$ примем минимальное из расстояний $l_{1,i,j}^{\alpha}$, $l_{2,i,j}^{\alpha}$:

$$l_{i,j}^{\alpha+1} = \min(l_{1,i,j}^{\alpha}, l_{2,i,j}^{\alpha}).$$

В качестве веса $v_{i,j}^{\alpha+1}$ логично принять сумму весов указанных ребер:

$$v_{i,j}^{\alpha+1} = v_{1,i,j}^{\alpha} + v_{2,i,j}^{\alpha}.$$

Таким образом, после завершения итераций оказываются полностью определенными топология графа G(T), а также веса его узлов w_i и значения атрибутов его ребер $(l_{i,j}, v_{i,j})$; $i, j \in [1 : n^T]$, $i \neq j$.

Вернемся к использованию в обозначениях весов узлов и атрибутов ребер графа G(T) индекса T .

Исключим из числа атрибутов ребер графа G(T) расстояния $l_{i,j}^T$ и модифицируем веса $v_{i,j}^T$ ребер этого графа: положим, что «новый» вес ребра (c_i, c_j) равен $v_{i,j}^T = v(l_{i,j}^T, v_{i,j}^T)$, где $v(l_{i,j}^T, v_{i,j}^T)$ – некоторая положительная убывающая функция расстояния $l_{i,j}^T$ и такая же возрастающая функция «старого» веса $v_{i,j}^T$. Например, можно принять:

$$v_{i,j}^T = \lambda_1 \frac{v_{i,j}^T}{l_{i,j}^T}. \quad (7)$$

При необходимости можно нормировать веса узлов и ребер полученного графа G(T), например, следующим образом:

$$w_i^T = \frac{w_i^T}{w_{\max}^T};\ v_{i,j}^T = \frac{v_{i,j}^T}{v_{\max}^T};\ i, j \in [1 : n^T];\ i \neq j.$$

Здесь $w_{\max} = \max_i w_i^T$, $v_{\max}^T = \max_{i,j} v_{i,j}^T$ – максимальный вес узла и ребра в графе G(T) соответственно.

Корпоративная база знаний представляет собой, как правило, совокупность разного рода слабоструктурированных документов, в которых с той или иной степенью подробности описаны прецеденты – ситуации и решения, которые были приняты в этих ситуациях. В системах поддержки принятия решений (СППР), которые используют такие базы знаний, поиск решения заключается в поиске в этих базах наиболее подходящих прецедентов и соответствующих им документов.

Выводы:

1. Рассматривается поиск решений по атрибутам документов, содержащимся в их метаданных, как альтернатива полнотекстовому поиску.
2. Рассматривается иной подход к поиску решений в базах знаний прецедентов, когда метаданные формируются на основе онтологии соответствующей предметной области, заданной в виде семантической сети. При этом релевантность документа оценивается близостью в некоторой метрике семантической сети этого документа и семантической сети запроса.
3. Предполагается, что по методике построения семантической сети документа построены семантические сети указанных кластеров. Таким образом, поисковые образы документа и запроса представляются в виде совокупности семантических сетей, соответствующих слотам паттерна проектирования и паттерна запроса.

В данной статье меры сложности строятся на основе таких параметров семантической сети, как количество входных и выходных понятий, реберная плотность и диаметр графа, соответствующего этой сети. Предложенная модель более строга, такая сеть классифицируется как однородная (с единственным типом отношений между понятиями) и бинарная (отношения связывают только по два понятия).

Литература

1. *И.П. Норенков.* Интеллектуальные технологии на базе онтологий // Информационные технологии, 2010, №1, С.17 − 23.
2. *А.П. Карпенко.* Меры важности концептов в семантической сети онтологической базы знаний [Электронный ресурс] // Наука и образование: электронное научно- техническое издание, 2010, 7. (http://technomag.edu.ru/doc/151142.html).
3. *М. Гринева, Д. Лизоркин*. Анализ текстовых документов для извлечения тематически сгруппированных ключевых терминов [Электронный ресурс]. (http://citforum.ru/database/articles/kw_extraction/).

4. *О.И. Ларичев*. Теория и методы принятия решений//Хроника событий в Волшебных странах. – М.: Университетская книга, Логос, 2006. −292 с.
5. *G.A. Miller and etc*. Wordnet: a lexical database for the english language [Электронный ресурс]. // (http://wordnet.princeton.edu/).
6. *Ю.А. Целых*. Теоретико-графовые методы анализа нечетких социальных сетей [Электронный ресурс]. (http://swsys.ru/print/article_print.php?id=742).