КИБЕРНЕТИКА

А.В. АНИСИМОВ, А.А. МАРЧЕНКО, Т.Г. ВОЗНЮК

УДК 681.3

ОПРЕДЕЛЕНИЕ СЕМАНТИЧЕСКИХ ВАЛЕНТНОСТЕЙ КОНЦЕПТОВ ОНТОЛОГИЙ С ПОМОЩЬЮ НЕОТРИЦАТЕЛЬНОЙ ФАКТОРИЗАЦИИ ТЕНЗОРОВ БОЛЬШИХ ТЕКСТОВЫХ КОРПУСОВ

Аннотация. Описан метод автоматического определения семантических отношений между концептами-узлами сети онтологической базы знаний на основе анализа матриц семантико-синтаксических валентностей слов. Данные матрицы получены с помощью неотрицательной факторизации тензоров синтаксической сочетаемости слов. Тензоры сгенерированы в процессе частотного анализа синтаксических структур предложений больших текстовых корпусов статей English Wikipedia и Simple English Wikipedia.

Ключевые слова: автоматическое извлечение знаний, корпусная лингвистика, онтологии, неотрицательная факторизация тензоров.

ВВЕДЕНИЕ

Неотрицательная тензорная факторизация (NTF) в последнее время является активно применяемой технологией в таких областях как информационный поиск, обработка изображений, обработка естественного языка, машинное обучение и в других смежных направлениях. Данный подход — один из наиболее перспективных для выявления и анализа взаимосвязей и отношений в массивах данных, где сочетаются объекты N разных типов и классов. В компьютерной лингвистике N-мерный тензор реализуется как многомерный массив данных, полученных при частотном анализе больших корпусов текстов. Тензор — удобная структура для представления данных высших порядков. Факторизация N-мерного тензора при ранге разложения k формирует N двумерных матриц, состоящих из k вектор-столбцов, представляющих отображение каждого отдельного измерения тензора на k факторизованных измерений латентного семантического пространства. Это служит уникальным средством для моделирования и выявления взаимосвязей лингвистических переменных в массиве N-мерных данных.

Факторизация тензоров — мультилинейный аналог сингулярного разложения матриц, используемого в латентном семантическом анализе для обработки двумерных массивов данных. Метод неотрицательной факторизации тензоров в некотором смысле можно назвать n-мерным обобщением латентного семантического анализа. Структуру, полученную в результате факторизации тензора, можно сравнить с многослойной нейронной сетью, состоящей из N слоев, которые представляют множества объектов N типов, и скрытого коммутационного слоя, состоящего из множества узлов с различными весовыми коэффициентами. Данный слой моделирует взаимосвязи между объектами N типов и связывает N слоев в единую нейронную сеть.

© А.В. Анисимов, А.А. Марченко, Т.Г. Вознюк, 2014

В настоящее время неотрицательная тензорная факторизация является перспективным методом решения задач компьютерной лингвистики, о чем свидетельствуют многочисленные работы в этом направлении [1–4].

Особый интерес представляют публикации [1, 2], в которых описываются модели тензорного представления данных о частоте различных типов синтаксических сочетаний слов в предложениях, например трехмерных сочетаний типа subject — verb — object или четырехмерных сочетаний типа subject — verb — direct_object — indirect_object, или других синтаксических сочетаний длины, не превышающей размерности тензора N. В тензоре каждому измерению соответствует некоторый фиксированный член предложения — подлежащее, сказуемое, дополнение, определение, обстоятельство и т.д. N-мерные тензоры содержат оценки частоты употребления в корпусах текстов сочетаний разных наборов слов в предложениях естественного языка. При этом учитываются синтаксические позиции слов. После обработки больших текстовых корпусов и накопления значительного объема данных в тензоре формируется N-мерный массив описания поведения лексических единиц в предложениях данного языка: т.е. для множества слов, описанных в тензоре, дано описание, в какие синтаксические отношения, с кем и с какой частотой слова имеют свойство вступать. Причем отношения эти не бинарные, а многомерные (N — максимальная размерность отношений). Затем следует этап неотрицательной факторизации полученного тензора. Факторизация приводит к значительному преобразованию модели представления данных. Изначально многомерный тензор разрежен и огромен по объему. Каждая из N осей синтаксического пространства содержит десятки тысяч или сотни тысяч точек-слов. После факторизации тензора его данные представляются в виде N двумерных матриц, состоящих из k-мерных векторов (где значение k намного меньше, чем количество точек-слов в любом из N измерений тензора). Параметр k — степень факторизации, размерность латентного семантического пространства, число признаковых измерений в нем. Помимо значительно более компактного и удобного представления массива данных предоставляется возможность быстрого вычисления оценки вероятности любого возможного сочетания слов в различных синтаксических конструкциях предложения. Это можно выполнить путем вычисления суммы произведений компонент N k-мерных векторов, соответствующих этим словам, выбранных из матриц, соответствующих их синтаксическим позициям. Например, чтобы проверить, насколько вероятно предложение «Повар жарит цыпленка», нужно найти в матрице SUBJECT k-мерный вектор s, который соответствует существительному «повар», затем найти в матрице VERB k-мерный вектор v, который соответствует глаголу «жарит». После этого найти в матрице DIRECT_OBJECT k-мерный вектор do, который соответствует существительному «цыпленок»; далее вычисляется сумма произведений соответствующих компонент этих трех векторов:

$$x_{svdo} = \sum_{i=1}^{k} s_i v_i do_i \quad (\text{для случая } N = 3),$$

где s_i — i-й элемент вектора s, v_i — i-й элемент вектора v, do_i — i-й элемент вектора do.

Если результат суммы превышает некоторый пороговый уровень, то делается вывод о существовании такой последовательности слов в предложении. Вычисление данной оценки для сочетания «Цыпленок жарит повара» приводит к выводу о невозможности существования такого варианта.

Данная модель позволяет весьма успешно автоматически выделять из корпусов текстов такие лингвистические структуры, как предпочтения сочетаемости

в предложениях (selectional preferences) [1] и субкатегориальные фреймы глаголов (Verb SubCategorization Frame) [2], которые сочетают в себе данные о семантических и синтаксических свойствах связей между глаголами и их аргументами-существительными в предложениях естественного языка. Из этого следует возможность автоматического выделения из полученного латентного семантического пространства семантических отношений типа семантических «ролевых» падежей Филмора [5]. Семантические ролевые падежи представляют систему разнотипных семантических связей между концептами-узлами, расположенными в иерархической сети некоторой онтологии, например лексико-семантической базы Wordnet [6]. Определение семантических связей между концептами онтологий в процессе обработки и анализа разложенных тензоров текстовых корпусов позволит автоматизировать наполнение контентом онтологических баз знаний.

Векторы из матриц разложенного тензора являются описанием частотного распределения лексем в последовательностях слов предложений.

Вектор из матрицы разложенного лингвистического тензора, принадлежащий некоторому слову, называем вектором семантико-синтаксической валентности данного слова в синтаксической позиции соответствующей матрицы.

Основная сложность состоит в том, что при построении тензора семантико-синтаксической сочетаемости лексем основными объектами изучения и анализа являются лексемы — слова, которые по природе неоднозначны. Векторное представление семантико-синтаксических валентностей любого слова W, определяемое соответствующим ему вектором из матрицы разложенного тензора, — это, по сути, сумма составляющих слагаемых векторов отдельных разных семантических значений этого слова W — концептов $Sw_1, Sw_2, \ldots, Sw_t$ в некоторой онтологии. В данной работе ставится задача — по вектору валентности $(v_1, v_2, \ldots, v_k)$ слова W получить составляющие слагаемые векторов валентностей

$$(v_{11}, v_{12}, \ldots, v_{1k}), (v_{21}, v_{22}, \ldots, v_{2k}), \ldots, (v_{t1}, v_{t2}, \ldots, v_{tk})$$

для каждого из его t значений. Вектор валентностей фиксированного значения — концепта онтологии — является неявным описанием его семантических отношений с другими концептами онтологической базы знаний.

В настоящей работе предложен новый метод определения семантических отношений между концептами — синсетами WordNet. Это реализуется посредством анализа разложенных тензоров, сформированных при обработке корпусов статей English Wikipedia [7] и Simple English Wikipedia [8], с расщеплением векторов семантической валентности слов на составляющие векторы семантической валентности их значений и с конкретной привязкой расщепленных векторов к соответствующим концептуальным узлам онтологии WordNet. Предложенный метод протестирован на точность разделения векторов семантической валентности (VSVs) слов на составные слагаемые **VSVs** концептов — значений данных слов, а также на точность их привязки к синсетам WordNet. Основное преимущество данного метода — полная автоматизация процесса нахождения новых семантических отношений между концептами семантической базы знаний в процессе анализа больших текстовых корпусов. Несмотря на то, что отношения задаются неявно — через векторы семантических валентностей, именно эта форма записи дает возможность решать такие классические задачи компьютерной лингвистики, как разрешение неоднозначности слов (word sense disambiguation), измерение семантической близости слов, семантический анализ текстов с использованием техники построения кратчайших расстояний в сети онтологии и многих других.

В настоящее время автоматическое извлечение лингвистических данных из текстовых корпусов — достаточно популярное направление компьютерной лингвистики. Создаются методы автоматизации получения различных типов информации, таких как селективные преференции [1], имена собственные [9], мультилингвистические связи [10], синтаксические правила [11–13], коллокации [14] и другие языковые структуры данных.

Современные методы автоматизации расширения и пополнения онтологий новыми знаниями о концептах и связях между ними можно разделить на следующие основные группы.

Методы, основанные на свойствах распределений слов. Этот подход заключается в изучении и выявлении данных о совместном распределении слов в текстах для того, чтобы вычислять семантическое расстояние между концептами, представленными этими словами. Построенная метрика может использоваться для кластеризации концептов [15], формального концептуального анализа [16], а также для классификации слов внутри существующих онтологий [17–20]. Данные методы используются для пополнения онтологий новыми концептами. Также актуальными являются работы по изучению отношений подчинения между словами в предложении, которые совместно с различными эвристиками дают возможность выделять не-таксономические отношения в онтологиях [21].

Методы, основанные на выделении и подборе шаблонов. Эти методы используют лексические и лексико-семантические шаблоны для выявления онтологических и не-таксономических отношений между концептами в текстах. В работах [22–24] вручную определяются регулярные выражения для выделения гипонимических и меронимических отношений. Число ошибок в работе данных методов составляет примерно 32 % [25]. Существуют также системы, которые комбинируют методы анализа распределений и шаблонный подход для выявления гипернимических и других не-таксономических отношений между концептами.

Методы, основанные на анализе текстов статей толковых словарей — тезаурусов — электронных энциклопедий. Данные методы [26–29] имеют то преимущество, что можно использовать стандартную структуру словарных статей, а также те отношения-связи, которые существуют между словарными статьями для организации структур данных в онтологиях. Определения концептов включают наиболее значимую информацию о них, а также основные связи с другими концептами, что является очень удобной нотацией для автоматического перевода в форму онтологической базы данных. В последнее время для работы в этом направлении в качестве основных ресурсов данных часто выбирают лексико-семантическую онтологию WordNet и электронную глобальную энциклопедию Wikipedia. Разработан ряд методов привязки статей Wikipedia в качестве новых концептуальных узлов к синсетам WordNet для увеличения покрытия семантического поля концептов в WordNet [30–32]. При этом решается основная задача нахождения в таксономии WordNet места для интеграции новых узлов, соответствующих статьям Wikipedia. Когда речь идет о семантически тождественных узлах Wikipedia и WordNet, задача решается относительно просто по сравнению со случаем, когда нужно интегрировать в иерархию WordNet узел статьи Wikipedia, который не имеет существующего аналога в WordNet. Тогда необходимо найти ближайший по смыслу узел WordNet и привязать новый узел к нему на правах семантического потомка.

Следующий важнейший этап — выделение не-таксономических отношений между концептами сети путем анализа текстов Wikipedia. В [31] предложен метод

выделения не-таксономических семантических отношений между концептами. Метод состоит в последовательной обработке Simple English Wikipedia, отборе всех входов (статей), после этого выполняется этап устранения неоднозначностей, после чего устанавливаются связи между соответствующими узлами-входами. Устранение неоднозначностей представляет собой четкую привязку каждого отдельного входа Simple English Wikipedia к конкретному синсету WordNet. Далее для каждого входа анализируется его определение и обнаруживаются слова, которые присутствуют в тексте определения. Если эти слова имеют связи-отношения со словом-входом в базе WordNet, то структура этих предложений анализируется и соответствующий этой паре слов синтаксический шаблон приписывается типу существующего в WordNet отношения. Далее синтаксические шаблоны, собранные на предыдущем этапе и приписанные некоторому типу отношений, сравниваются, и те из них, которые имеют схожую структуру и конфигурацию, автоматически обобщаются. Обобщенные шаблоны в дальнейшем используются для выделения из текстов Wikipedia межпонятийных отношений, которые не были прописаны в WordNet до того и впоследствии после обнаружения добавляются в базу. Таким образом, определено более 2600 новых отношений, которых изначально в WordNet не было. Точность определения таких отношений, зависящая от типов отношений и степени обобщения, выбранной для шаблонов, лежит в пределах 60–70 % для лучших комбинаций.

В работе [32] описан метод привязки статей Wikipedia к синсетам WordNet. Для устранения неоднозначности слов-входов Wikipedia использовался метод Personalized PageRank с регулируемым пороговым уровнем. При этом привязка на тестовой выборке достигает оценки точности 0.78 по мере F1.

Предлагаемый в настоящей статье алгоритм отличается от вышеупомянутых методов тем, что использует неотрицательную факторизацию многомерного тензора, содержащего данные о сочетаемости слов языка на разных синтаксических позициях в предложениях текстов, взятых из корпуса статей Wikipedia. Этот метод применен для создания векторного пространства описания семантических валентностей данных слов. После этого векторы семантических валентностей слов раскладываются на векторы семантических валентностей отдельных значений этих слов с привязкой к соответствующим им синсетам WordNet. Таким образом формируется неявное описание семантических отношений между концептуальными узлами WordNet. Для решения этой конкретной задачи неотрицательная факторизация лингвистических тензоров до настоящего времени не применялась. Сравнение представленного метода с какими-либо другими существующими подходами в рамках данной статьи представляется сложной задачей, так как методы привязки, например статей Wikipedia к узлам WordNet, применяют принципиально другие подходы и постановка задачи абсолютно другая. Простое сравнение цифр напротив оценок точности «привязки» заведомо не является корректным, ввиду совершенно другой постановки задачи привязки к синсетам WordNet не статей Wikipedia, а расщепленных векторов семантической валентности для всех упомянутых в корпусе значений некоторого слова. Неотрицательная тензорная факторизация широко применяется для решения других задач, не являющихся алгоритмически подобными описываемой в статье. Поэтому сравнение по линии использования алгоритмов неотрицательной тензорной факторизации может быть проведено преимущественно по временным характеристикам, что не является целью данной работы.

МЕТОДИКА СБОРКИ ТЕНЗОРА ТЕКСТОВОГО КОРПУСА

Для сборки N-мерных тензоров текстовых корпусов в качестве основной использовалась методика, представленная в работе [2]. При этом некоторые детали алгоритма были модифицированы в силу специфики поставленной задачи по изучению и извлечению именно семантических отношений между концептами — узлами онтологической сети базы знаний.

В начальной фазе корпусы проходят этап синтаксического анализа предложений текстов, который проводится с помощью стендфордского парсера — Stanford Parser [33].

Далее постсинтаксический анализатор, разбирая синтаксическое дерево, выделяет главный глагол предложения — на него указывает root (ROOT-0, ***verb***); субъект-подлежащее — nsubj (verb, ***noun***); прямой объект-дополнение — dobj (verb, ***noun***); непрямой объект — iobj (verb, ***noun***); существительное в предложной группе — prep_during (verb, ***noun***), prep_on (verb, ***noun***), prep_in (verb, ***noun***) и т.д.; межглагольную связку xcomp(verb, ***verb*1**). Таким образом, при анализе предложения, находя лексемы в соответствующих синтаксических позициях, система заполняет этими словами кортеж предложения (root-verb, nsubj, dobj, iobj, prep_, xcomp, count), при этом в prep_ существительное записывается вместе с предлогом. Если в предложении отсутствует некоторая синтаксическая позиция, то она заполняется символом пустого слова $\varnothing$. В тензор помещаются только кортежи как минимум с тремя ненулевыми полями. В count сохраняется число раз использования подобного лексического сочетания в данном корпусе. Шесть первых элементов кортежей формируют координаты пространства, седьмой — значения частоты сочетаний. Как результат, формируется 6-мерный тензор лексических сочетаний в данных синтаксических позициях. Отдельного рассмотрения заслуживает вопрос определения лексем — заполнителей полей кортежа. В большинстве случаев все сводится к задаче анализа группы существительного в конкретной синтаксической позиции и определения того слова или словосочетания, которое будет корректным заполнителем поля кортежа. В случае сложной группы существительного определяется, не включает ли в себя данная группа существительного в позиции головы некоторое имя статьи Wikipedia (например, ***Серная кислота*** или ***Эйфелева башня***). Если включает, то заполнителем является найденное имя статьи Wikipedia, если нет — то голова данной группы существительного.

В [2] используется более сложная синтаксическая модель сборки тензора с большим числом синтаксических шаблонов связей и, как результат, формируется тензор с размерностями 9–12. В настоящей работе не ставилась задача построить максимально сложную структуру. Приоритетом было — повышение вероятности правильного определения синтаксических отношений и понижение уровня шума. Кроме того, цель данной публикации — разработка методов определения семантической связи некоторой размерности между концептами — синсетами WordNet, и на данном этапе не стоит задача исчерпывающего обнаружения всех семантических отношений, описанных в корпусе. Предпочтение отдано точности определения найденных семантических отношений. В любом случае сложность синтаксической модели и увеличение размерности тензора являются вопросами технической реализации, и они будут увеличиваться в дальнейших исследованиях.

РАЗЛОЖЕНИЕ ТЕНЗОРА

Под факторизацией N-мерного тензора T в данной статье понимается построение его представления в виде суммы внешних произведений N векторов. В линейной алгебре внешнее произведение обычно является обозначением тензорного произведения двух векторов. Результат применения внешнего произведения к паре векторов — матрица. Например, внешнее произведение четырехмерного вектора u и трехмерного вектора v задается матрицей

$$u \circ v^{\mathrm{T}} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} [v_1 \; v_2 \; v_3] = \begin{bmatrix} u_1 v_1 & u_1 v_2 & u_1 v_3 \\ u_2 v_1 & u_2 v_2 & u_2 v_3 \\ u_3 v_1 & u_3 v_2 & u_3 v_3 \\ u_4 v_1 & u_4 v_2 & u_4 v_3 \end{bmatrix}.$$

Внешнее произведение трех векторов даст, таким образом, трехмерную матрицу — 3-мерный тензор.

В данном случае нужно построить представление $T = \sum_{i=1}^{k} x_{1i} \circ x_{2i} \circ \ldots \circ x_{ti}$, которое максимально близко по значениям всех элементов к первоначальным значениям в тензоре T. После этого k векторов x_{1i} собираются в матрицу X_1, k векторов x_{2i} — в матрицу X_2 и т.д., k векторов x_{ti} — в матрицу X_t. Каждая из них будет состоять из векторов размерности k.

Описанный в предыдущем разделе тензор размерности 6 должен быть представлен в виде суммы k внешних произведений шестерок векторов. Векторы из этой суммы должны складываться в виде шести соответствующих матриц, каждая из которых будет представлять отображение множества лексем, стоящих в определенной синтаксической позиции, на множество k фактор-измерений латентного семантического пространства семантико-синтаксических отношений слов текстового корпуса.

Для разложения тензора используется метод неотрицательной тензорной факторизации. Он подобен параллельному факторному анализу с ограничением, что все данные должны быть неотрицательными. Параллельный факторный анализ — это мультилинейный аналог сингулярного разложения матриц, используемого в латентном семантическом анализе [34]. Главная суть метода — минимизация суммы квадратов разности между оригинальным тензором и его факторизированной моделью. Для N-арного тензора $T \in R^{D_1 \times D_2 \times \ldots \times D_N}$ определяется целевая функция

$$\min_{x_{1i} \in R^{D_1}, x_{2i} \in R^{D_2}, \ldots, x_{Ni} \in R^{D_N}} \left\| T - \sum_{i=1}^{k} x_{1i} \circ x_{2i} \circ \ldots \circ x_{Ni} \right\|_F^2, \qquad (1)$$

где k — размерность факторизированой модели, $\circ$ — внешнее произведение.

Для неотрицательной факторизации добавляются ограничения по неотрицательности значений элементов:

$$\min_{x_{1i} \in R_{\geq 0}^{D_1}, x_{2i} \in R_{\geq 0}^{D_2}, \ldots, x_{Ni} \in R_{\geq 0}^{D_N}} \left\| T - \sum_{i=1}^{k} x_{1i} \circ x_{2i} \circ \ldots \circ x_{Ni} \right\|_F^2. \qquad (2)$$

Результат работы алгоритма — представление тензора в виде N матриц, которые описывают отображение каждой из размерностей тензора на k фактор-измерений латентного семантического пространства. Обычно NTF-модель подгоняется методом наименьших квадратов. На каждой итерации $N-1$ размерностей фиксируется, а N-я размерность подгоняется методом наименьших квадратов. Процесс продолжается до момента сходимости. Число фактор-измерений латентного семантического пространства было взято $k = 150$. В [2] экспериментально установлено, что именно это значение обеспечивает лучшие результаты факторизации. Для решения задачи неотрицательной факторизации 6-мерного тензора корпуса текстов статей Wikipedia применялся алгоритм параллельной факторизации PARAFAC [35]. Для разложения тензора разработана собственная программная реализация алгоритма. При этом удалось достичь значительного ускорения процесса решения задачи благодаря распараллеливанию вычислений на графической карте по технологии, аналогичной описанной в [36].

Пример генерации матриц факторизированного тензора текстового корпуса. Дан текстовый корпус:

«Мама мыла раму. Витя любит футбол. Юля любит цветы. Маша слушает оперу. Юля делает уроки. Мама посадила цветы. Маша любит оперу. Мама поливает цветы. Юля смотрит футбол. Юля любит цветы. Юля любит футбол. Маша любит цветы. Витя любит оперу. Мама наняла повара. Повар жарит цып-

ленка. Мама наняла повара. Цыпленок клюет зерно. Мама наняла повара. Цыпленок клюет цыпленка».

Каждому слову внутри каждой синтаксической категории присваиваем уникальный номер *Id*, соответствующий его координате на соответствующей оси трехмерного пространства тензора.

Subject (1 — Мама, 2 — Витя, 3 — Маша, 4 — Юля, 5 — Повар, 6 — Цыпленок)

Predicate (1 — мыла, 2 — любит, 3 — слушает, 4 — делает, 5 — посадила, 6 — поливает, 7 — смотрит, 8 — жарит, 9 — клюет, 10 — наняла)

Object (1 — раму, 2 — футбол, 3 — оперу, 4 — уроки, 5 — цветы, 6 — цыпленка, 7 — зерно, 8 — повара)

В результате разбора текста генерируется 3-мерный тензор со следующими ненулевыми элементами:

$$T[1,1,1]=1,\ T[2,2,2]=1,\ T[4,2,5]=2,\ T[3,3,3]=1,\ T[4,4,4]=1,\ T[1,5,5]=1,$$

$$T[3,2,3]=1,\ T[1\,6,5]=1,\ T[4,7,2]=1,\ T[4,2,2]=1,\ T[4,2,5]=1,\ T[3,2,5]=1,$$

$$T[2,2,3]=1,\ T[1,10,8]=3,\ T[5,8,6]=1,\ T[6,9,7]=1,\ T[6,9,6]=1.$$

Все остальные элементы тензора равны нулю, поэтому получен разреженный тензор.

В результате проведения неотрицательной факторизации данного тензора получено его разложение на сумму произведений троек векторов ($k=11$, именно это значение оказалось оптимальным — оно максимально точно приближает модель к первоначальным значениям входного тензора T):

$$T=\sum_{i=1}^{11}\lambda_i x_i\circ y_i\circ z_i=\lambda_1 x_1\circ y_1\circ z_1+\lambda_2 x_2\circ y_2\circ z_2+\ldots+\lambda_{11}x_{11}\circ y_{11}\circ z_{11},$$

где λ_i — весовой коэффициент i-го фактор-измерения.

В данном примере алгоритм вычислил значения $\lambda_1=2$ и $\lambda_9=3$, все остальные $\lambda_i=1$ для $i\neq 1$ и $i\neq 9$.

После этого векторы $x_1, x_2, \ldots, x_{11}$ складываются вместе в матрицу:

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}
Мама	0.0	0.0	1.0	1.0	0.0	1.0	0.0	0.0	1.0	0.0	0.0
Витя	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
Маша	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Юля	1.0	0.0	0.0	0.0	1.0	0.0	1.0	0.0	0.0	0.0	0.0
Повар	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0
Цыпленок	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0.

Векторы $y_1, y_2, \ldots, y_{11}$ складываются вместе в матрицу:

	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	y_9	y_{10}	y_{11}
Мыла	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Любит	1.0	1.0	0.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0
Слушает	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Делает	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0
Посадила	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Поливает	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0
Смотрит	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0
Жарит	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0
Клюет	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0
Наняла	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0.

Векторы $z_1, z_2, \ldots, z_{11}$ складываются вместе в матрицу:

	z_1	z_2	z_3	z_4	z_5	z_6	z_7	z_8	z_9	z_{10}	z_{11}
Раму	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Футбол	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0
Оперу	0.0	1.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
Уроки	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0
Цветы	1.0	0.0	0.0	1.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0
Цыпленка	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	1.0
Зерно	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0
Повара	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0.

Допустим, требуется получить корпусную частотную оценку для словосочетания «Повар жарит цыпленка».

Из матрицы Подлежащих X берется вектор семантико-синтаксической валентности слова «Повар»:

Повар (0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 1.0 0.0).

Из матрицы Сказуемых Y берется вектор семантико-синтаксической валентности слова «жарит»:

Жарит (0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 1.0 0.0).

Из Дополнений Z берется вектор семантико-синтаксической валентности слова «цыпленка»:

Цыпленка (0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 1.0 1.0).

Вычисляя по формуле $E = \sum_{i=1}^{11} \lambda_i x_i y_i z_i$, получаем частотную оценку для словосочетания $E = 1$ (пересечение ненулевых значений только по десятой координате). Она свидетельствует о том, что подобное предложение содержится в текстовом корпусе один раз. Это означает, что подобное лексико-семантическое отношение (повар — подлежащее, жарит — сказуемое, цыпленок — дополнение) является допустимым, т.е. соответствует языковой практике. Для предложения «Цыпленок жарит повара» частотная оценка $E = \sum_{i=1}^{11} \lambda_i x_i y_i z_i = 0$, что говорит об отсутствии описания подобного отношения во входном корпусе. Легко проверить, что таким же образом вычисляются все первоначальные значения тензора T входного текстового корпуса.

Описанный метод факторизации текстовых тензоров эффективен на больших текстовых корпусах широкого тематического охвата, когда в своих векторах матрицы разложенного тензора в неявном виде содержат информацию о миллионах и миллиардах семантических отношений между десятками и сотнями тысяч слов. Полученные матрицы тензорной факторизации представляют собой ценный источник информации для решения различных задач лингвистического анализа и формируют базу данных лексико-семантических отношений между словами естественного языка.

АЛГОРИТМ РАСЩЕПЛЕНИЯ ВЕКТОРОВ СЕМАНТИКО-СИНТАКСИЧЕСКОЙ ВАЛЕНТНОСТИ СЛОВ НА СОСТАВЛЯЮЩИЕ СЛАГАЕМЫЕ ВЕКТОРЫ ВАЛЕНТНОСТЕЙ ИХ РАЗНЫХ ЗНАЧЕНИЙ

После факторизации построенного 6-мерного тензора корпуса статей Wikipedia было получено шесть матриц: ROOT_VERB, NSUBJ, DOBJ, IOBJ, PREP_, XCOMP, которые состоят из векторов размерности k ($k = 150$). Каж-

дый вектор-столбец этих матриц соответствует некоторому слову или словосочетанию. Данные векторы описывают семантико-синтаксическое поведение слов, а именно в каких синтаксических позициях какие связи и с кем некоторое слово образует. По аналогии с химической терминологией данные векторы называем векторами семантико-синтаксических валентностей (**VSV**) слов. Слова по своей природе неоднозначны, т.е. им, как правило, соответствует несколько значений. Таким образом, вектор слова является суммой векторов всех значений данного слова. Одному слову может соответствовать несколько векторов из разных матриц, соответствующих разным синтаксическим позициям, задача расщепления каждого из этих векторов решается отдельно. Разработанный алгоритм расщепления **VSV** слова на множество **VSVs** всех его значений — синсетов WordNet — имеет следующий вид:

дан вектор семантической валентности V, размерности k, который соответствует некоторому слову w в NSUBJ (в любой из шести матриц метод работает аналогично). Существительному w соответствует t значений — синсетов в WordNet. Требуется разбить V на составляющие слагаемые $V_1, V_2, \ldots, V_t$, соответствующие данным t синсетам.

Алгоритм задается следующим образом:

for $i = 1$ to k do
begin
if $V[i] \neq 0$ then
for $j = 1$ to t do
begin

{Необходимо определить, какому из t синсетов принадлежит i-е значение $V[i]$. Оно может принадлежать либо одному из синсетов, либо нескольким — тогда нужно разложить $V[i]$ на сумму $V_1[i] + V_2[i] + \ldots + V_t[i]$}

$S = 0$;

Взять j-й синсет и записать в r число слов в нем;

For p:= 1 to r do

begin

взять из NSUBJ вектор W, который соответствует p-му слову j-го синсета;

$S = S + W[i]$;

End;

$S_{\text{mean}} = S / r$;{вычислили средний коэффициент привязки $V[i]$ к словам j-го синсета}

Получить непосредственного предка j-го синсета — Anc и аналогично S_{mean} вычислить средний коэффициент привязки $V[i]$ к его словам $S_{\text{Anc-mean}}$;

Получить множество потомков j-го синсета на единичном расстоянии — {Offs} и аналогично S_{mean} вычислить средний коэффициент привязки $V[i]$ к каждому потомку $S_{\text{Offs-mean}}$ и выбрать среди них максимальный $S_{\text{MaxOffs-mean}}$;

Вычислить суммарную оценку для j-го синсета

$S_{\text{Fin}}[j] = C_1 * S_{\text{mean}} + C_2 * S_{\text{Anc-mean}} + C_3 * S_{\text{MaxOffs-mean}}$;

{C_1, C_2, C_3 — коэффициенты приоритетов, выбранные эмпирически}

End;

$V_1[i], V_2[i], \ldots, V_t[i]$ определяются из системы уравнений:

1. $V_1[i] = S_{\text{Fin}}[1] * X$; $V_2[i] = S_{\text{Fin}}[2] * X$; …; $V_t[i] = S_{\text{Fin}}[t] * X$;

2. $\sum_{j=1}^{t} S_{\text{Fin}}[j] * X = V[i]$; Определить значение $X = \dfrac{V[i]}{\sum_{j=1}^{t} S_{\text{Fin}}[j]}$;

For $j = 1$ to t do if $S_{\text{Fin}}[j] < R$ then $V_j[i] = 0$ else $V_j[i] = S_{\text{Fin}}[j] * X$;

{R — пороговый уровень, подобранный экспериментально}

end;

ЭКСПЕРИМЕНТАЛЬНЫЕ РЕЗУЛЬТАТЫ

В начале экспериментов был произведен подбор оптимальных коэффициентов C_1, C_2 и C_3 из вышеописанного алгоритма. Их первые приблизительные значения были присвоены эмпирически. Далее алгоритмом была обработана выборка из приблизительно 3600 **VSVs** слов, равномерно выбранных из всех матриц факторизованного тензора, сформированного в процессе обработки текстов корпуса статей Wikipedia (приблизительно по 600 **VSVs** из каждой матрицы). Следует отметить, что при формировании данной выборки обеспечивалось обязательное выполнение условия: для каждого слова из выборки гарантированно существовал как минимум один вход — синсет WordNet. После обработки алгоритмом была получена выборка **VSVs** концептов с привязкой к конкретным синсетам WordNet для оценки корректности расщепления **VSV** слова на множество **VSVs** всех его различных значений. Для автоматизации процесса оценивания точности расщепления векторов слов и привязки составных слагаемых векторов к концептам — синсетам WordNet, была разработана программа. По набору полученных **VSVs** синсетов WordNet она генерирует множество всех возможных последовательностей слов — предложений с их участием, согласуемых со значениями в этих k-мерных **VSVs** синсетов. Далее эксперты с помощью этой же программы провели анализ корректности сформированных словосочетаний с исправлением ошибок путем замены неверно выбранных синсетов в сгенерированных предложениях на правильные синсеты в местах ошибок. Таким образом была сформирована учебная выборка, которая позволила подобрать оптимальные коэффициенты C_1, C_2 и C_3 для алгоритма расщепления и привязки **VSVs**.

Была решена оптимизационная задача подбора C_1, C_2 и C_3 для максимизации корреляции составленного множества возможных последовательностей слов в предложениях, соответствующих значениям в **VSVs**, полученных алгоритмом расщепления и привязки, к составленной экспертами учебной выборке корректных последовательностей слов в предложениях. Для решения задачи использовался метод имитации отжига [37] — вероятностная эвристика для решения задач глобальной оптимизации. В качестве функции для максимизации применен коэффициент корреляции Спирмена.

После подбора оптимальных значений C_1, C_2 и C_3 алгоритмом были расщеплены и привязаны к синсетам WordNet **VSVs** из матриц, полученных при неотрицательной факторизации тензоров корпусов статей Wikipedia и Simple Wikipedia. Для оценки качества работы алгоритма аналогичным образом, как и на этапе формирования обучающей выборки, была сформирована тестовая выборка **VSVs** концептов — синсетов Wordnet (при этом она полностью отличалась по составу от обучающей выборки). Выборка насчитывает примерно 5400 нормально распределенных по WordNet векторов, привязанных к синсетам — примерно по 450 векторов из каждой матрицы по каждому из корпусов. С помощью описанной выше программы генерации последовательностей слов, «разрешенных» в **VSVs**, была оценена точность (семантическое соответствие) привязки **VSVs** концептов к конкретным синсетам WordNet. Оценки точности работы алгоритма расщепления векторов валентности слов на составные слагаемые векторы их разных значений и привязки их к синсетам WordNet представлены в табл. 1.

Таблица 1

Матрицы	Wikipedia	Simple Wikipedia
VERB	79.84	73.54
NSUBJ	87.17	81.21
DOBJ	85.62	80.17
IOBJ	86.08	79.09
PREP_	83.45	75.61
XCOMP	73.91	69.08

Анализ данных, полученных при тестировании, четко выявляет несколько тенденций в оценках точности разработанного алгоритма. Во-первых, заметно понижение точности при расщеплении **VSVs** глаголов (матрицы VERB и XCOMP) по сравнению с точностью работы алгоритма с матрицами имен существительных

(NSUBJ, DOBJ, IOBJ, PREP_). Данное понижение точности алгоритма при работе с векторами глаголов объясняется самой природой глаголов. Глаголы — относительно малочисленный класс лексики по сравнению с именами существительными. Кроме того, глаголы в среднем имеют гораздо больше значений на одну лексему по сравнению с теми же существительными. Таким образом, задача расщепления **VSVs** глаголов с дальнейшей привязкой к синсетам WordNet объективно является значительно более сложной задачей. Ее решение требует анализа и обработки текстовых корпусов значительно большего объема (и по количеству текстов, и по охвату разнообразных тематик), чем это нужно для работы алгоритма с именами существительными. Также можно видеть относительно небольшое уменьшение оценок точности работы алгоритма при обработке матрицы PREP_ — матрицы валентностей существительных в составе предложных групп. В эту матрицу записывались существительные со всеми предлогами, соединенные в пары: ***at_University, in_Sweden, to_Granada***. Это многократно увеличивает число точек по данной шкале предложных групп с именами существительными. Такое увеличение тензорного пространства по одному из измерений также требует большего объема текстовых корпусов для достаточного равномерного заполнения тензора. Это отражается на некотором снижении точности работы алгоритма, которое, однако, не превышает 4–5 %, что можно считать вполне приемлемым результатом.

Более высокие оценки точности работы алгоритма расщепления **VSV** слов на составные слагаемые векторы их разных значений и привязки их к синсетам WordNet при работе с матрицами разложенного тензора корпуса текстов статей Wikipedia по сравнению с оценками работы алгоритма с тензорными матрицами корпуса статей Simple Wikipedia объясняются, безусловно, значительно большим объемом первого корпуса как по количеству статей (4,1 млн статей Wikipedia против 98 тыс. статей Simple Wikipedia), так и по их объему, где также значительный перевес имеет Wikipedia. Простые синтаксические структуры Simple Wikipedia обеспечивают почти 100 % качества при синтаксическом анализе, и, как следствие, достаточно высокое качество сборки тензора. Поэтому, несмотря на тотальный перевес в объеме Wikipedia по сравнению с Simple Wikipedia, отставание в оценках качества работы алгоритма на разложенном тензоре Simple Wikipedia не превышает в среднем 6–7 %. В целом представленные оценки свидетельствуют о достаточно высокой эффективности предложенного алгоритма и о серьезных перспективах его использования на практике.

ЗАКЛЮЧЕНИЕ

Данная работа описывает модель неотрицательной факторизации N-мерных лингвистических тензоров, собранных в процессе частотного анализа синтаксических структур предложений в больших текстовых корпусах. Разложение собранных тензоров в виде N матриц сокращенной размерности k, помимо компактной и удобной структуры представления данных о сочетаемости последовательностей лексем в некоторых синтаксических позициях предложений естественного языка, дает эффективный метод вычисления оценки вероятности существования семантико-синтаксических связей между словами разных грамматических категорий. При этом можно рассматривать k-мерные векторы из матриц факторизованного тензора как векторы семантико-синтаксических валентностей (**VSV**) слов. Так как слова по своей природе неоднозначны, и одному слову, как правило, соответствует несколько значений, в работе предложено рассматривать k-мерные **VSVs** слов как суммы составных слагаемых **VSVs** разных значений этих слов. В статье представлен разработанный метод расщепления **VSVs** слов на составные слагаемые **VSVs** их разных значений и привязки этих расщепленных составных слагаемых **VSVs** к синсетам WordNet в качестве их собственных значений **VSVs**. Алгоритм использует данные о синонимии слов в синсетах WordNet, а также гипонимические-гипернимические связи синсетов в иерархии онтологической сети. Реализованный алгоритм протестирован

рядом экспериментов с матрицами разложенных тензоров корпусов текстов статей Wikipedia и Simple Wikipedia. Полученные при тестировании оценки точности работы алгоритма демонстрируют его высокую эффективность.

Необходимо подчеркнуть, что значительное преимущество предложенного метода состоит в высокой степени автоматизируемости каждого этапа его работы — парсинг статей, сборка тензора, неотрицательная тензорная факторизация, расщепление векторов семантической валентности слов на векторы их значений и привязка к соответствующим синсетам WordNet. Все этапы выполняются полностью без участия человека. Эксперты необходимы только на этапе настройки алгоритма расщепления и привязки. Одним из актуальных направлений дальнейших исследований является максимальная минимизация участия экспертов в настройке алгоритма.

Задание семантических отношений между концептами-узлами онтологического графа в неявном виде с помощью k-мерных векторов семантических валентностей также имеет неоспоримое преимущество универсальности представления семантических связей. При обнаружении $(n+1)$-го типа связи между существующими концептами система фиксирует в базе его наличие в векторном представлении, не выдвигая немедленного требования пополнения списка отношений новым типом, его полного описания в онтологии и указания для него соответствующего синтаксического шаблона.

Данные преимущества метода указывают на реальные перспективы его использования на практике в автоматизации методов наполнения контентом онтологических баз знаний для автоматического определения семантических отношений между концептами — узлами онтологической сети в процессе обработки больших текстовых корпусов.

СПИСОК ЛИТЕРАТУРЫ

1. Van de Cruys T. A Non-negative tensor factorization model for selectional preference induction // J. Natural Language Engineer. — 2010. — **16**, N 4. — P. 417–437.
2. Van de Cruys T., Rimell L., PoibeauT., Korhonen A. Multi-way tensor factorization for unsupervised lexical acquisition // Proceedings of COLING 2012. — Mumbai, India — P. 2703–2720.
3. Cohen S.B., Collins M. Tensor decomposition for fast parsing with latent-variable PCFGs // NIPS. — 2012. — P. 2528–2536.
4. Peng W., Li T. On the equivalence between nonnegative tensor factorization and tensorial probabilistic latent semantic analysis // Appl. Intel., Springer J. — October 2011. — **35**, N 2. — P. 285–295.
5. FIllmore C.J. The case for case // E. Bach, R. Harms (Eds): Universals in Linguistic Theory. — New York: Holt, Rinehart, and Winston, 1968. — P. 1–88.
6. Introduction to WordNet: An on-line lexical database / G. MIller, R. Beckwith, C. Fellbaum, D. Gross, K. Miller. — http://wordnetcode.princeton.edu/5papers.pdf.
7. http://en.wikipedia.org/wiki/Main_Page.
8. http://simple.wikipedia.org/wiki/Main_Page.
9. Mikheev A., Grover C., Moens M. Description of the ltg system used for muc-7 // Proc. of 7th Message Understanding Conference (MUC-7). — 1998. — P. 1–12.
10. Dagan I., Itai A., Schwall U. Two languages are more informative than one // Proc. of ACL-91. — Berkeley, California. — 1991. — P. 130–137.
11. Hockenmaier J., Bierner G., Baldridge J. Providing robustness for a ccg system // Proc. of the Workshop on Linguist. Theory and Grammar Implement. — Birmingham. — 2000. — P. 97–112.
12. Briscoe T., Carroll J. Automatic extraction of subcategorization from corpora // Proc. of the 5th Conf. on Appl. Natural Language Proces. (ANLP-97), Washington DC, USA, 1997.
13. Xia F. Extracting tree adjoining grammars from bracketed corpora // Fifth Natural Language Proces. Pacific Rim Symp. — (NLPRS-99). — Beijing, China. — 1999.

14. Church K., Gale W., Hanks P., Hindle D. Using statistics in lexical analysis // U. Zernik (ed.) Lexical Acquisition: Exploiting On-line Resources to Build a Lexicon. — Hillsdale, New Jersey: Lawrence Erlbaum Associates, 1991. — Ch. 6. — P. 115–164.
15. Lee L. Similarity-based approaches to natural language processing // Ph.D. thesis. Harvard University Techn. Rep. TR-11-97. — 1997. — http:/www.cs.cornell.edu/home/llee/papers/thesis.pdf.
16. Cimiano P., Staab S. Clustering concept hierarchies from text // Proc. of LREC. — 2004. — P. 1721–1724.
17. Hastings P.M. Automatic acquisition of word meaning from context, University of Michigan // Ph.D. Dissertation. — 1994. — http://reed.cs.depaul.edu/peterh/papers/hastingsdiss.pdf.
18. Hahn U., Schnattinger K. Towards text knowledge engineering // AAAI/IAAI. — 1998. — P. 524–531. — URL citeseer.nj.nec.com/43410.html.
19. Pekar V., Staab S. Word classification based on combined measures of distributional and semantic similarity// Proc. of Research Notes of the 10th Conf. of the European Chapter of the Assoc. for Comput. Linguistics, Budapest. — 2003. — P. 147–150.
20. Alfonseca E., Manandhar S. Extending a lexical ontology by a combination of distributional semantics signatures // Knowledge Engineering and Knowledge Management. — Lecture Notes in Artificial Intelligence. — 2002. — **2473**. — P. 1–7.
21. Maedche A., Staab S. Discovering conceptual relations from text // Proc. of the 14th Europ. Conf. on Artificial Intel. — 2000. — P. 1–17.
22. Hearst M.A. Automatic acquisition of hyponyms from large text corpora // Proceedings of COLING-92. — Nantes, France. — 1992. — P. 539–545.
23. Hearst M.A. Automated discovery of WordNet relations // Ch. Fellbaum (Ed.) WordNet: An Electronic Lexical Database. — MIT Press, 1998. — P. 132–152.
24. Berland M., Charniak E. Finding parts in very large corpora // Proc. of ACL-99. — 1999.
25. Kietz J., Maedche A., Volz R. A method for semi-automatic ontology acquisition from a corporate intranet // Workshop “Ontologies and text”, co-located with EKAW’2000. — Juan-les-Pins, French Riviera. — 2000. — P. 2–6.
26. Providing machine tractable dictionary tools / Y. Wilks, D.C. Fass, C.M. Guo, J.E. McDonald, T. Plate, B.M. Slator // J. of Comput. and Translat. 2. — P. 99–154.
27. Rigau G. Automatic acquisition of lexical knowledge from MRDs // PhD Thesis, Departament de Llenguatges i Sistemes Inform’atics, Universitat Polit’ecnica de Catalunya. — 1998. — URL http:// adimen.si.ehu.es/~rigau/publications/thesis-rigau.pdf.
28. Richardson S.D., Dolan W.B., Vanderwende L. MindNet: acquiring and structuring semantic information from text // Proc. of COLINGACL’98, Montreal, Canada. — 1998. — **2**. — P. 1098–1102.
29. Dolan W., Vanderwende L., Richardson S.D. Automatically deriving structured knowledge bases rfon on-line dictionaries // PACLING 93 Pacific Association for Comput. Linguistics. — 1993. — P. 5–14.
30. Ruiz-Casado M., Alfonseca E., Castells P. Automaticassignment of Wikipedia encyclopedic entries to WordNet synsets // Advances inWeb Intelligence. — Berlin; Heidelberg: Springer, 2005. — P. 380–386.
31. Ruiz-Casado M., Alfonseca E., Castells P. Automatising the learning of lexical patterns: An application to the enrichment of Wordnet by extracting semantic relationships from Wikipedia // Data & Knowledge Engineering. — 2007. — **61**, Issue 3. — P. 484–499.
32. Niemann E., Gurevych I. The people’s web meets linguistic knowledge: Automatic sense alignment of Wikipedia and WordNet // Proc. of the 9th Intern. Conf. on Comput. Semantics (IWCS). — 2011. — P. 205–214.
33. http://nlp.stanford.edu/software/lex-parser.shtml.
34. Indexing by latent semantic analysis / S. Deerwester, S.T. Dumais, G.W. Furnas, T.K. Landauer, R. Harshman // J. of the American Soc. for Inform. Sci. — 1990. — **41**, N 6. — P. 391–407.
35. Harshman R. Foundations of the PARAFAC procedure: models and conditions for an explanatory multimodal factor analysis // UCLA Working Papers in Phonetics. — 1970. — **16**. — P. 1–84.
36. Nonnegative tensor factorization accelerated using GPGPU / J. Antikainen, J. Havel, R. Josth, A. Herout, P. Zemcik , M. Hauta-Kasari // IEEE Trans. Parallel Distrib. Syst. — 2011. — **22**, N 7. — P. 1135–1141.
37. Kirkpatrick S., Gelatt C.D., Vecchi M.P. Optimization by simulated annealing // Sci. 220. — 1983. — P. 671–680.

Поступила 25.07.2013