Построение концептуальных графов как элементов семантической разметки текстов[1]

Creating conceptual graphs as elements of semantic texts labeling

Богатырёв М. Ю. (okkambo@mail.ru), **Тюхтин В. В.**

Тульский государственный университет

В работе рассматриваются возможности применения концептуальных графов в качестве средства семантической разметки корпусов текстов. Такая разметка образует метаданные, позволяющие эффективно решать некоторые задачи Text Mining. Предлагается алгоритм автоматического построения концептуальных графов, приводятся результаты экспериментов на текстах аннотаций научных статей.

1. Введение

Концептуальные графы являются одной из семантических моделей текста, относящейся к классу семантических сетей. Впервые концептуальные графы были предложены в работах Дж. Совы, обобщенные результаты которых представлены в его монографии [1], и в настоящее время играют важную роль как средство моделирования *структур, наделенных смыслом*, в таких областях как математическая лингвистика, биоинформатика, математическая логика.

Концептуальный граф — это двудольный направленный граф, состоящий из двух типов узлов: *концептов* и *концептуальных отношений*, или просто *отношений*. Для концептуальных графов разработан стандарт их представления (см. также работу Дж. Совы [2]) и языки описания, среди которых наиболее популярны CGIF (Conceptual Graph Interchange Form) и XML — представление концептуальных графов.

Концептуальные графы применяются в задачах анализа текстовой информации, относящихся к направлению, обозначаемому термином *Text Mining*. В одной из последних обзорных российских работ М.С. Куприянова и др. [3] термин Text Mining именно так и переводится: «*анализ текстовой информации*». В англоязычной литературе Text Mining — это разновидность анализа данных *Data Mining* (иногда называемая text Data Mining), причем существенная особенность анализа в обоих вариантах состоит в том, что он направлен на *извлечение знаний* из данных, текстовых или иных. Направления Text Mining и Data Mining объединяет другое направление — *поиск знаний в базах данных* (Knowledge Discovery in Databases, KDD), которое в последнее время чаще называют *поиск знаний в данных,* используя ту же аббревиатуру. Для краткости, и учитывая неоднозначность трактовки, мы будем употреблять термин Text Mining без перевода.

В задачах Text Mining строятся *кластеры*, *ассоциации*, анализируются *особенности* текстов, *подобие* текстов и т.д. с целью извлечения знаний из текстовых данных. При этом термин «знание» трактуются как некоторая «овеществленная» модель знаний: процесс извлечения знаний приводит к нахождению конкретных значений параметров заранее заданной модели знаний. Все модели знаний условно можно разделить на два класса: модели в виде правил и модели в виде структур. Структурные модели образуют иерархию. Вершину ее составляют модели в виде формальных онтологий, а элементарной структурной моделью, соответствующей каждому предложению текста, является концептуальный граф.

Формально семантика концептуальных графов задается логическими выражениями, формируемыми на графе. Соответственно, логика предикатов первого порядка остается к настоящему времени основным математическим аппаратом исследования концептуальных графов. Здесь получено много результатов, касающихся фундаментальных свойств концептуальных графов. Стиль работ данного направления хорошо иллюстрирует работа [4].

[1] Работа выполнена при поддержке РФФИ, грант № 07-07-00276-а.

Традиционные методы обработки текстов используют ключевые слова или векторные модели текста, что требует значительных вычислительных ресурсов при обработке больших текстовых коллекций. Концептуальный граф как модель сложнее, чем набор ключевых слов, но компактнее, чем, например, вектор, построенный на тексте в методах латентно — семантического анализа. Это позволяет эффективно применять концептуальные графы в задачах Text Mining. Примером является работа [5], где на концептуальных графах решаются задачи кластеризации и выявляются отклонения сравниваемых текстов. В работе [11] мы применили концептуальные графы к построению ассоциативных правил, извлекаемых из текста.

Несмотря на признание концептуальных графов в качестве семантической модели текста и постоянный интерес к ним (см., например, электронный ресурс [12]), практическое применение концептуальных графов требует решения ряда проблем. Среди них одной из важнейших является проблема автоматизации построения концептуальных графов. Другой проблемой является поддержка концептуальных графов в реальных системах Text Mining. Появление размеченных корпусов текстов как элементов информационных систем открывает здесь новые направления исследований и, по-видимому, позволит получать более эффективные решения задач Text Mining

В данной работе представлены некоторые результаты, относящиеся к алгоритмизации построения концептуальных графов и их поддержке в пилотном исследовательском проекте электронной библиотеки.

2. Автоматическое построение концептуальных графов

Автоматизация построения концептуальных графов является в целом нерешенной проблемой. Сложность данной проблемы обусловлена смысловым многообразием, присутствующем в любом тексте естественного языка. Поэтому каждому предложению текста может соответствовать несколько концептуальных графов. Данный факт приводит к идее связать построение концептуальных графов с решаемыми при их помощи задачами. Другими словами, строить графы под задачу. На практике так и происходит: из всего многообразия задач автоматизации построения концептуальных графов решают некоторое подмножество задач, актуальных с точки зрения конечной цели применения концептуальных графов. Данный принцип применялся нами. Например, в рассматриваемом ниже алгоритме из всех знаков пунктуации анализируются только запятые.

Существует несколько подходов к построению концептуальных графов. Согласно *вербоцентрическому* подходу, в каждом предложении фиксируется центральный концепт — глагол, задающий главный смысл предложения; детализация смысла задается другими концептами и отношениями. При этом необходимо обеспечить корректную работу алгоритма в достаточно сложных случаях — в предложениях без глаголов или имеющих сложные глагольные формы. Другим важным известным решением является применение *семантических ролей для* построения отношений, которое рассмотрено ниже.

Известны системы, использующие автоматическое построение концептуальных графов для англоязычных текстов. В работе Ангеловой и др. [6] изложены принципы такого построения на основе вербоцентрического подхода. Развитием данной работы является работа S. Hensman [7], где для построения концептуальных графов привлекаются известные ресурсы VerbNert и WordNet.

Нами разрабатывается подобная система для англо — и русскоязычных текстов. В системе применяется алгоритм построения концептуальных графов, главные пункты которого состоят в следующем.

1. Анализ языка предъявленного текста; выбор между русским и английским языками. Данный этап необходим, ввиду принципиальных различий в обработке англо — и русскоязычных текстов алгоритмом.
2. Разделение предъявленного текста на предложения. Разделение предложений на слова, знаки пунктуации и иные символы.
3. Определение синтаксических элементов предложения. Построение нормальных форм для синтаксических единиц.
4. Определение морфологических признаков элементов предложения.
5. Формирование концептов из списка элементов предложения. В качестве концептов выбираются основные части речи, исключая частицы, союзы, предлоги, вводные слова.
6. Определение концептуальных отношений и акторов. *Актором* называется внешнее отношение, смысл которого назначается, а не извлекается из предложения.

Реализация пунктов 1, 2 не вызывает проблем. Синтаксический анализ в п.3 выполняется на основе известных алгоритмических решений АОТ [13]. Как показали эксперименты, данных решений недостаточно и в алгоритм добавлены значительное число как синтаксических правил, так и новых, в ряде случаев эвристических решений.

Морфологический анализ в п.4 выполняется с привлечением системы DWARF [14] и ее словарных ресурсов.

Самой сложной задачей при построении концептуальных графов является задача построения отношений. Известны подходы к решению данной

задачи, основанные на *разметке семантических ролей* предложения [8]. В нашей системе развивается подобный подход для русскоязычных текстов, основанный на применении шаблонов.

2.1. Выделение семантических ролей

Семантической ролью называется совокупность черт, общих для одинаково кодируемых элементов предложения. Сложность разметки семантических ролей обусловлена тем, что роль может не совпадать с элементом предложения и определяться не одним, а несколькими элементами. Разметка семантических ролей требует не только синтаксического, но и морфологического анализа текста.

При выделении семантических ролей применяются четыре множества объектов:

- множество атрибутов элементов предложения для русского языка,
- множество шаблонов в виде правил,
- множество атрибутов шаблонов и
- множество семантических ролей.

Шаблон содержит список специальных атрибутов, характеризующих сочетания слов анализируемого предложения. Шаблон определяет семантическую роль, но не тождественен ей. Среди атрибутов шаблонов имеются, например, такие: *название связи*, *тип связи*, *направление поиска главного слова* и т.д.

Имеется несколько типов шаблонов:

- *двухуровневый шаблон* — проверяет два рядом стоящих элемента предложения;
- *трёхуровневый шаблон* — проверяет три рядом стоящих элемента предложения;
- *грамматический шаблон* — проверяет три элемента и дополнительно наличие конца или начала предложения или соседнего знака пунктуации.

Множество шаблонов составляется на основе правил русского языка, но может пополняться новыми шаблонами с учетом конкретной лексики, например, научной.

Множество семантических ролей состоит из имен и описаний семантических ролей. В настоящее время в системе применяются роли: *агенс*, *пациенс*, *генетив*, *реципиент, атрибут, модификатор, объект, тема, источник, цель.*

После выполнения синтаксического разбора алгоритм получает список элементов предложения и каждому элементу соответствует код, собранный на множестве атрибутов элементов предложения для русского языка. Данный код используется далее при подборе шаблона, обрабатывающего элементы предложения. Последовательной проверкой соответствия шаблонов сочетаниям элементов предложения и применением соответствующих правил шаблонов на списке элементов предложения алгоритм добивается полного разбора предложения. Примененные для обработки словосочетаний шаблоны порождают семантические роли, которые тождественны концептуальным отношениям. При этом словосочетания, прошедшие обработку каждым шаблоном правил, удаляются и далее не рассматриваются. Поэтому порядок применения шаблонов имеет существенное значение и, как показывают эксперименты, влияет на результаты построения концептуальных отношений.

Например, применение одного из правил АОТ слева направо и справа налево дает различные результаты, причем правильным является второй результат, показанный на рис. 1б.

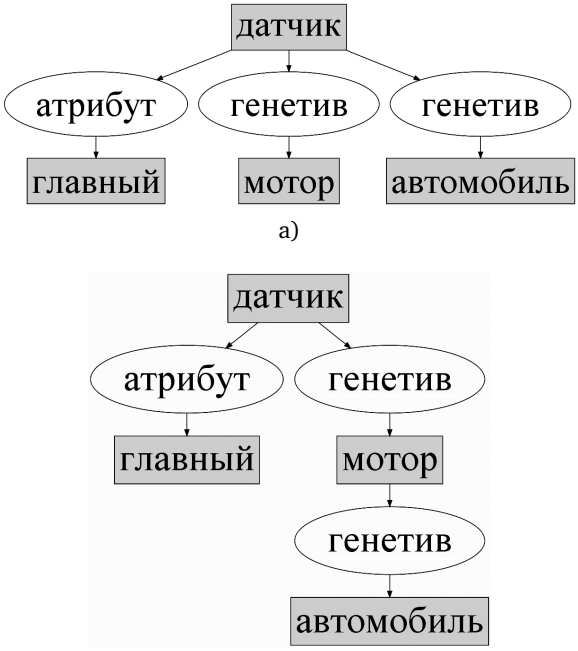

Рис. 1. Варианты концептуальных графов при разборе фразы «главный датчик мотора автомобиля» слева направо (а) и справа налево (б).

В стандартном варианте применения правила на рис. 1-а концепты *главный*, *мотор*, *автомобиль* равноправны, а в инверсном варианте на рис. 1б имеет место иерархия концептов, которая позволяет правильно интерпретировать смысл фразы «главный датчик мотора автомобиля»: мотор принадлежит автомобилю.

При определении морфологических признаков в алгоритме применяется ряд новых правил, позволяющих строить концептуальные графы более корректно. Так к правилу обработки однородных прилагательных добавлено правило построения отношения «*модификатор*» между прилагательным и местоимением в дательном падеже. Результат иллюстрируется примером на рис.2.

На рис. 2 показаны примеры моделирования фразы «предоставить необходимую им информацию» до введения правила построения отношения «модификатор» (рис. 2а) и после введения правила (рис. 2б).

На рис. 2-а показаны «бездомные» концепты, не связанные никакими отношениями, и выявляемые системой при интерпретации графов. Добавление отношения «модификатор» исправляет ошибку.

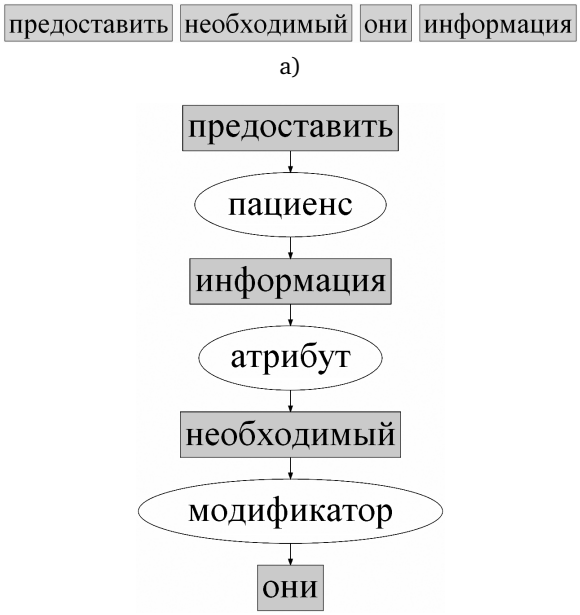

Рис. 2. Варианты моделирования фразы «предоставить необходимую им информацию».

Появление «бездомных» концептов связано с важной проблемой регулярности алгоритма.

2.2. Проблема регулярности.

Рассмотренный здесь алгоритм является сходящимся, то есть всегда приводит к построению концептуального графа. Однако, важной алгоритмической проблемой, возникающей при построении подобных систем, остается *проблема регулярности*. Нестрого, суть проблемы сводится к следующему: гарантирует ли алгоритм однотипные, регулярные решения на множестве данных, также являющихся однотипными, или отличающихся от однотипных применением к ним регулярных правил? Другими словами, построив при помощи алгоритма правильный концептуальный граф для одного предложения, вправе ли мы ожидать, что для другого предложения концептуальный граф будет построен так же корректно?

Данная проблема для задач Text Mining общего алгоритмического решения не имеет.

В нашем случае признаком нерегулярности алгоритма является появление «бездомных» концептов, показанных на рис. 2а. Увеличение числа правил в алгоритме приводит к исключению «бездомных» концептов, как это видно из рис. 2б, однако проблема регулярности остается.

С целью исследования данной проблемы в системе используется контролируемое увеличение числа правил алгоритма. В результате в интерфейсе системы введена дополнительная опция выбора «*глубины смысла концептуального графа*», получаемого из предложения. Данная опция имеет численное значение. При установке опции в наибольшее её значение мы получаем наиболее полный результирующий граф, а при установке опции в наименьшее значение (0), получаем граф, содержащий основной (вербальный) смысл предложения. «Глубина смысла» имеет также промежуточные значения. При увеличении значения опции в граф добавляются различные типы связей, причём таким образом, что значимость добавляемых связей обратно пропорциональна уровню глубины смысла. То есть, чем глубже исследуется смысл, тем более незначительные для основного смысла предложения элементы и связи добавляются в концептуальный граф.

Отметим одно полезное свойство алгоритма, связанное с проблемой регулярности. При разборе предложений с ошибками, включающими несогласованные, посторонние элементы, алгоритм порождает «бездомные» концепты. В этом случае их индикация позволяет выявить необычные особенности текста.

3. Применение концептуальных графов в проекте электронной библиотеки

Рассмотренный алгоритм реализован в исследовательском проекте электронной библиотеки, ориентированной на хранение текстов научных публикаций. Отметим две специфические функции, которые планируется реализовать в проекте дополнительно к стандартным библиотечным функциям.

1. **Диагностика новой информации, появляющейся в библиотеке**. Данная функция необходима, когда ресурсы библиотеки пополняются из сети Интернет. При реализации функции требуется решение задачи Text Mining, известной как *извлечение фактов — Fact Extraction*. На концептуальных графах данная задача может быть решена методом выявления отклонений в сравниваемых текстах [5].
2. **Концептуальная трассировка**. Используя данную функцию, пользователь системы, введя текст — запрос, получает в качестве выходных данных иерархическую структуру понятий, *имеющих отношение* к данному запросу. Функция полезна для обучения. Ее реализация требует построение онтологий.

Тексты научных публикаций загружаются в систему из сети Интернет, из открытого источника [15]. Практически все тексты данного источника англоязычные. Для проведения экспериментов на русскоя-

зычных текстах используется источник [16] — труды конференций RCDL за 10 лет в объеме 546 единиц.

Концептуальные графы строятся только для аннотаций статей, поскольку аннотации статей призваны сжато и точно отражать их содержание. Кроме того, аннотации имеют ограниченную лексику, что важно для повышения регулярности работы алгоритма построения концептуальных графов. Построенные графы обрабатываются подсистемой интерпретации. Среди ее функций разрабатывается функция принятия решения о загрузке текста статьи в библиотеку по результатам анализа ее аннотации. Здесь как раз необходима диагностика новой информации.

Реализация рассмотренных функций основана на решении задач *агрегирования* и *кластеризации* на концептуальных графах, однако не сводится только к ним. Рассмотрение указанных задач выходит за рамки данной работы, их более детальное описание можно найти в работе [9].

Построение концептуальных графов выполняет специальная подсистема. В ней имеется диалоговый режим и режим автоматического построения концептуальных графов по находящимся в базе данных текстам аннотаций.

В диалоговом режиме реализовано полное управление процессом построения концептуальных графов: можно корректировать результаты работы алгоритма (с заданием упомянутой выше «*глубины смысла концептуального графа*»), изменять и вводить новые концепты и отношения, а также акторы, использовать визуализацию (см. рис. 1, 2), удалять графы, конвертировать их в разные форматы.

В автоматическом режиме соответствующие аннотациям генерируемые графы пополняют базу данных графов в формате XML. При этом происходит отбраковывание «неправильных» графов, имеющих «бездомные» концепты (см. выше). Также особо фиксируются несвязные графы.

Сравнение работы алгоритма на русскоязычных и англоязычных текстах дало, как и ожидалось, значительно большее число «неправильных» графов для русскоязычных текстов, чем для англоязычных. Это объясняется двумя причинами:

- известной сложностью русского языка по сравнению с английским — большим числом правил и их нерегулярностью;
- несоблюдением принципа компактности в русскоязычных аннотациях — наличием длинных и очень длинных предложений, что повышает вероятность генерации «неправильных» графов.

3.1. Поддержка контекстов.

Главной проблемой построения концептуальных графов, которая решается в настоящее время, является *поддержка контекстов*.

Каждый концептуальный граф строится для одного предложения, но предложения могут быть связаны по смыслу. Часто страдательный залог, употребленный в предложении, является индикатором важной ссылки на внешнюю информацию — *контекст*, содержащуюся в других предложения или вне анализируемого текста. Построение концептуальных графов для предложений в страдательном залоге является в настоящее время самой трудной задачей. Однако, получаемые здесь «неправильные» графы могут быть полезны в задаче поддержки контекстов.

Проиллюстрируем данную ситуацию следующим примером. На рис. 3 показан концептуальный граф, соответствующий предложению «Много статей посвящено данной теме».

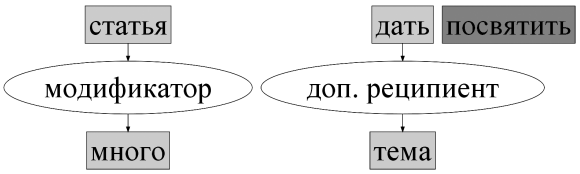

Рис. 3. Концептуальный граф предложения «Много статей посвящено данной теме»

Полученный граф является несвязным, что будет отслежено подсистемой построения графов. Отношение *дополнительный реципиент* как раз и введено для применения в подграфах несвязных графов. Правильно ли построен данный граф? С одной стороны, страдательный залог, имеющийся в предложении, никак не отражен. С другой стороны, отдельный подграф [*дать*] → (*доп. реципиент*) → [*тема*] задает особенность, которую мы можем дальше интерпретировать. Информация (знание), представляемая данным графом, сводится к следующему:

- существует много статей на некоторую тему;
- эта тема не описана в данном предложении, но, возможно, описание дано в других предложениях.

Таким образом, тема в анализируемом предложении определена в контексте. Мы можем сопоставить с графом рис. 3 индикатор: «Дать тему!», определив его буквально как правый подграф графа. Данный индикатор может служить признаком контекста и инициировать его построение.

Формализм концептуальных графов определяет контекст как *концепт графа, с которым связан некоторый непустой другой граф* [2]. В нашем примере концепт *тема* и является контекстом. Остается построить по остальному тексту граф (его может не быть!), описывающий тему, которой посвящено много статей.

Рассмотренный пример является частным случаем, индуцирующим практически важную, на наш взгляд, задачу: имея концептуальные графы как структуры микро-уровня (уровня одного предложения), строить онтологии как структуры макро-

уровня. Строя таким образом онтологии *снизу вверх*, в отличие от применяемых сейчас в системах построения онтологий методов *сверху вниз*, мы обеспечим большую адекватность онтологии той информации, которая содержится в соответствующих ей данных. Построение онтологий данным способом при помощи концептуальных графов связано с решением двух задач — задачи агрегирования концептуальных графов и задачи поддержки контекстов на концептуальных графах.

3.2. Концептуальные графы как элементы разметки корпусов текстов

Идеология разрабатываемой в данном проекте системы соответствует идеологии *корпусов текстов*. В самом деле, корпус текстов — это способ компьютерного хранения текстовых данных нового поколения, который, кроме собственно текстов, содержит их *разметку*. Разметка представляет собой *метаданные*, отражающие как лингвистическую, так и экстралингвистическую информацию, касающуюся хранимых текстов.

Разметка определяется задачами, решаемыми на текстах корпуса. Примерами двух принципиально разных классов задач, решаемых на корпусах, являются *лингвистические исследования текстов* и *извлечение знаний из текстов корпусов*. Последний класс задач соответствует назначению разрабатываемой в данном проекте системы.

Семантической разметкой назовем отображение текста корпуса на некоторую семантическую модель, например, концептуальный граф. Элементы текста — слова и предложения — отображаются в элементы модели — концепты, и отношения концептуального графа.

Применение концептуальных графов в качестве полноценного средства разметки текстов неразрывно связано с организацией корпуса как информационной системы, что сводится к следующему:

- автоматизация построения концептуальных графов;
- организация хранения концептуальных графов;
- алгоритмическая и вычислительная поддержка задач, решаемых при помощи концептуальных графов.

Данные элементы являются составными частями рассмотренной здесь системы. Ее развитие может быть связано с решением задач корпусной лингвистики.

4. Выводы и дальнейшие исследования

Автоматическое построение концептуальных графов как семантических моделей текста является алгоритмически сложной задачей. Применение для ее решения разметки семантических ролей предложения позволяет получить приемлемые результаты. Однако, данного решения недостаточно для построения графов для предложений в страдательном залоге. Поддержка контекстов является возможным решением в данном случае, не выводящим за пределы формализма концептуальных графов.

Кроме того, построенные концепты и соответствующие им графы — контексты могут служить основой для построения онтологий. Такой способ построения онтологий *снизу вверх*, по-видимому, будет способствовать сохранению той информации, которая содержится в порождающих онтологии данных. Однако, данная идея требует более тщательной проработки.

Дальнейшие исследования по данной теме планируется выполнить в следующих направлениях:

- исследование и доработка алгоритма построения концептуальных графов в части реализации поддержки контекстов;
- переход к задаче агрегирования концептуальных графов с последующей разработкой инструментов интерпретации графов — агрегатов как элементов онтологий;
- выполнение вычислительных экспериментов с алгоритмом построения концептуальных графов в режиме реального времени.

Литература

1. *Sowa, J. F.* Knowledge Representation: Logical, Philosophical, and Computational Foundations, Brooks Cole Publishing Co., Pacific Grove, CA. 2000.
2. *Sowa J.F. Conceptual Graphs: Draft Proposed American National Standard, //International Conference on Conceptual Structures ICCS-99, Lecture Notes in Artificial Intelligence 1640, Springer 1999.*
3. *Технологии* анализа данных: Data Mining, Visual Mining, Text Mining, OLAP/ А. А. Барсегян, М. С. Куприянов, В. В. Степаненко, И. И. Холод. — СПб: БХВ-Петербург, 2008, — 384 с.
4. *Chein M., Mugnier Marie-Laure.* Conceptual Graphs: fundamental notions // Revue d'Intelligence Artificielle, Vol. 6, n 4, 1992, pp 365–406.
5. *Montes-y-Gomez, M., Gelbukh, A., Lopez-Lopez, M.* Text Mining at Detail Level Using Conceptual Graphs. //Lecture Notes In Computer Science; Vol. 2393. P. 122–136.
6. *Boytcheva, S. Dobrev, P. Angelova, G.* CGExtract: Towards Extraction of Conceptual Graphs from Controlled English. //Lecture Notes in Computer Science № 2120, Springer 2001.
7. *Hensman, S.* Construction of conceptual graph representation of texts. In Proceedings of Student Research Workshop at HLT-NAACL, Boston, 2004, p.p. 49–54.
8. *Gildea D., Jurafsky D.* Automatic labeling of semantic roles //Computational Linguistics, 2002, v. 28, p.p. 245–288.
9. *Богатырев М. Ю., Латов В. Е., Столбовская И. А., Тюхтин В. В.* Эволюционный подход к задаче кластеризации на концептуальных графах и его применение в системах поддержки электронных библиотек. — Математические методы распознавания образов. 13 Всероссийская конференция. Сб. докладов. — М.: МАКС Пресс, 2007. — 668 с. — С. 464–468.
10. *Богатырев М. Ю., Латов В. Е., Столбовская И. А.* Применение концептуальных графов в системах поддержки электронных библиотек. // Электронные библиотеки: перспективные методы и технологии, электронные коллекции. Труды Девятой Всероссийской научной конференции RCDL'2007 (Переславль-Залесский, Россия, 15–18 октября 2007). — Т. 2, С. 104–110.
11. *Богатырев М. Ю., Тюхтин В. В.* Решение некоторых задач Text Mining при помощи концептуальных графов. // Электронные библиотеки: перспективные методы и технологии, электронные коллекции. Труды Десятой Всероссийской научной конференции RCDL'2008 — Дубна, 2008. — 415 с. — С. 31- 36.
12. *Электронный* ресурс: A World of Conceptual Graphs: http://conceptualgraphs.org/
13. *Электронный* ресурс: Автоматическая Обработка Текста http://aot.ru/
14. *Электронный* ресурс: Cognitive Technologies — Интеллектуальные технологии управления. http://www.cognitive.ru
15. *Электронный* ресурс: Scientific Literature Digital Library http://citeseer.ist.psu.edu/
16. *Электронный* ресурс: Труды конференций Электронные библиотеки: Перспективные Методы и Технологии, Электронные коллекции http://rcdl.ru/