УДК 681.035.01

ОБЗОР МЕТОДОВ ПОИСКА РЕЛЕВАНТНЫХ ДОКУМЕНТОВ

ЗИМИН И.В., ГЛУШКОВА И.И.
izvestiya@ktu.aknet.kg

Рассмотрен обзор методов поиска релевантных документов. Информационно-поисковые системы предназначены для решения более общей задачи поиска, чем поиск на точное соответствие. Конечной целью поиска является выбор релевантной поисковому запросу информации, степень релевантности которой можно определить как степень её смысловой близости к поисковому запросу. При этом поиск информации по смыслу рассматривает возможность наряду с построением семантических сетей документов применение онтологических знаний предметных областей.

История развития человечества указывает, что для своего дальнейшего движения вперед необходимо накопление информации в различных ее видах и формах, ее обработка и передача следующим поколениям. С появлением электронных носителей информации увеличилась пользовательская потребность в быстром и релевантном поиске информации в постоянно растущем объёме информационных ресурсов, имеющих в своем большинстве вид текстовых документов.

Существующие информационные системы, работающие с электронными текстовыми документами, можно разделить на две категории [1]: информационно-поисковые системы (в зарубежной терминологии они фигурируют под термином information retrieval systems) и системы выборки данных (data retrieval systems). При этом отмечают, что данная классификация условна, и в её контексте многие современные информационные системы совмещают в себе свойства как систем выборки данных, так и информационно-поисковых систем (ИПС). Базовые отличия систем выборки данных и информационно-поисковых систем представлены в таблице 1.

Таблица 1

Сравнительный анализ систем выборки данных и ИПС.

Критерии оценки	Системы выборки данных	Информационно-поисковые системы
Соответствие данных поисковому запросу	точное	частичное
Классификация документов	детерминированная	вероятностная
Язык запросов	искусственный	естественный
Критерии выборки документов	булева функция релевантности	вероятностная функция релевантности
Устойчивость к ошибкам в данных и запросах	неустойчивы	устойчивы

Типичным примером систем выборки данных являются классические реляционные СУБД, где в качестве языка запросов используется тот или иной диалект языка запросов SQL. Информационно-поисковые системы предназначены для решения более общей задачи поиска, чем поиск на точное соответствие. Конечной целью поиска является выбор релевантной поисковому запросу информации, степень релевантности которой можно определить как степень её смысловой близости к поисковому запросу. И системы выборки данных, и информационно - поисковые системы работают с некоторой коллекцией документов. Исходную коллекцию документов можно рассматривать как список записей (документов), где каждая запись содержит в себе некоторый список слов, состоящих из символов алфавита. В реальных информационно-поисковых системах в исходном множестве документов содержится дополнительная информация, описывающая документы, которая также используется для осуществления поиска. Рассмотрим этот класс информационных систем более подробно.

Основные понятия. ***(Информационно-поисковые системы).*** Поисковая система в общем виде представлена на рис. 1. В зависимости от типа задачи пользователь формирует различные запросы и обращается к различным поисковым механизмам. Однако

общим для всех задач является то, что:

1) пользователь формирует запрос (A) естественным для него (пользователя) способом;
2) пользователь желает искать среди объектов (C), представленных в традиционном для него виде;
3) машина, получая запрос пользователя, преобразовывает его к формальному описанию (B), затем ищет среди коллекции документов, заранее преобразованных к формальному виду (D), наиболее близкие к запросу пользователя (релевантные) документы.

Под формальным описанием исходных объектов поиска, или образами объектов, понимается представление объекта (документа, запроса) в виде списка его признаков, например, слов или словосочетаний, сопровождаемого информацией о значимости (весе) каждого признака для содержания конкретного документа. Процесс предварительной обработки документов, нацеленный на формирование образов документов, называют индексацией коллекции документов. В рамках данной работы интерес представляет поисковый механизм F, который в известных поисковых системах зачастую остается скрытым для изучения.

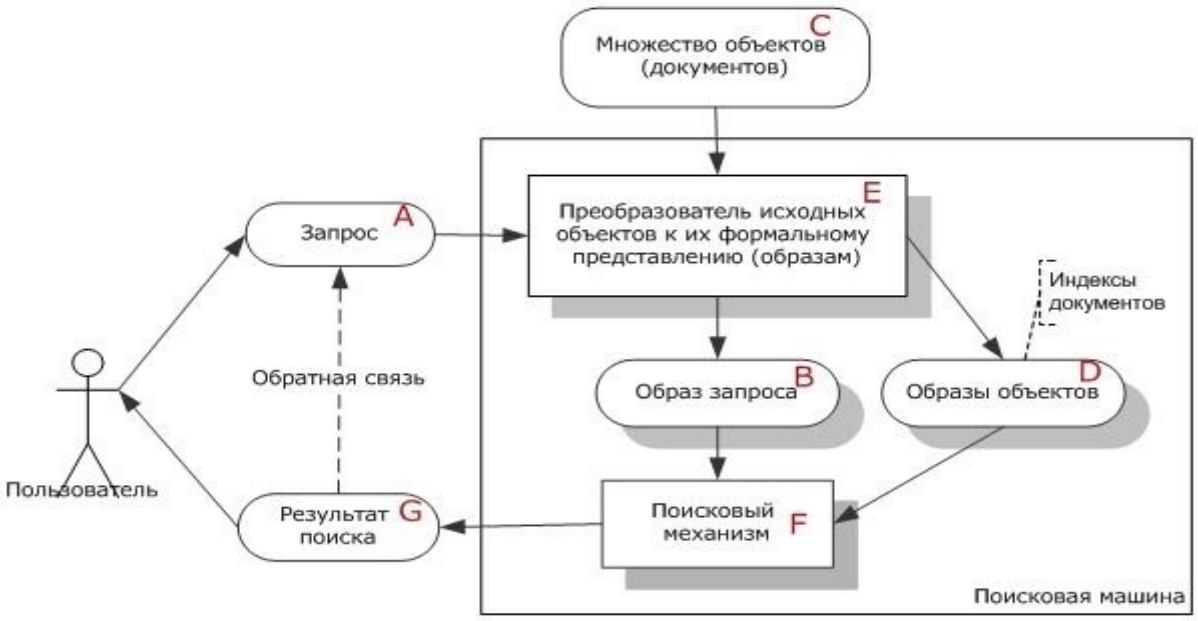

Рис. 1. Поисковая система в общем виде

Классификация поисковых методов. Поисковые методы, заложенные в основу поискового механизма F (рис. 1), можно классифицировать по различным основаниям. Рассмотрим некоторые из них.

По ***виду поиска*** можно выделить 3 группы:

1) полнотекстовый поиск — поиск по всему содержимому документа. Как правило, полнотекстовый поиск для ускорения поиска использует предварительно построенные индексы. Наиболее распространенной технологией для индексов полнотекстового поиска являются инвертированные индексы;
2) поиск по метаданным — это поиск по неким атрибутам документа, поддерживаемым системой. Такими атрибутами могут выступить название документа, дата создания, размер, автор и т. д.;
3) поиск по изображению — поиск по содержанию изображения. Для организации такого вида поиска поисковая система распознает содержание фотографии или картинки. В результатах поиска пользователь получает похожие изображения.

По ***типу решаемых задач*** информационный поиск можно разделить на 2 класса:

1) поиск текстовых ресурсов, содержащих информацию по запросу пользователя; характерен для процесса разыскивания пользователем хоть какой-либо информации, касающейся интересующей тематики. Он используется в случае, если пользователь незнаком с предметной областью (и, соответственно, не может сформулировать точный запрос для получения точного ответа) или пользователь не нашел в базе проиндексированных текстов ответ на свой вопрос и пытается найти хоть какую-нибудь информацию по данной тематике;
2) поиск смысловой информации, содержащей ответ на вопросы пользователя или информацию, содержательно соответствующую запросу пользователя; характерен для отыскания конкретной

информации пользователем, знакомым с прикладной областью и уверенным, что искомая информация может содержаться в проиндексированных текстах.

По ***типу определения соответствия*** поисковые методы делятся на:

1) булевый поиск, где запросы строятся на основе элементарных терминов документов (слов или словоформ), находящихся между собой в отношениях, определённых предикатами, такими как дизъюнкция, конъюнкция и отрицание. Булевые запросы обычно используются в поиске на точное соответствие, в котором документы проверяются на наличие того или иного термина;
2) поиск по релевантности, где осуществляется полная и точная выборка подмножества документов, наиболее близких по смыслу (релевантных) поисковому запросу. Здесь под понятием близости документа поисковому запросу понимают некоторую функцию корреляции между каждым документом и запросом, вычисление которой позволяет определить степень близости информационного содержания документа и запроса. Вид функции релевантности зависит от конкретной реализации информационно-поисковых систем. В свою очередь функция релевантности документа запросу может вычисляться:
   2.1) на основе вероятности, базирующейся на некоторой информации о документах, автоматически получаемой системой в ходе анализа их информационного контекста документов (вплоть до анализа синтаксиса и семантики);
   2.2) с использованием методов статистического анализа текста документов [2]. В данном контексте основной проблемой информационно-поисковых систем является автоматическая обработка и классификация документов для последующего вычисления функции релевантности, где в качестве языка запросов может использоваться язык запросов, приближённый к естественному;
3) поиск по сходству представляет собой модификацию булевого поиска, где поисковой системой учитываются возможные неточности в задании поисковых терминов или в электронном представлении документов. Поиск по сходству может быть очень полезен при выборке информации в специализированных электронных библиотеках, таких как медицинские базы данных, или в системах распознавания текста, где существует определённая вероятность неточного задания термина в поисковом запросе или же в самом документе. Меру корреляции между терминами поискового запроса и терминами исходных документов обычно определяют в виде расстояния редактирования [3, 4].

По ***принципу информационного поиска*** поисковые методы делятся на:

1) методы поиска на основе кластерных методов и векторных моделей;
2) методы поиска по ключевым словам (терминам) документов [5].

Рассмотрим кратко наиболее распространенные методы, нашедшие применение в поисковых механизмах различных ИПС.

Постановка задачи и область исследования. ***Кластерные методы*** позволяют реализовать процесс классификации исходного множества документов или их составных элементов, классифицировать документы на кластеры, в которых все элементы по определённому набору их свойств можно считать. Многие методы классификации элементов основаны на парных отношениях между элементами, подлежащими классификации. Таким образом, разбив все документы на кластеры и определив в каждой группе характерного представителя: центроид кластера, можно сравнивать запрос не с каждым документом по отдельности, а сначала только с центроидами. Если центроид релевантен запросу, то поиск продолжают внутри кластера, если нет - перейти к рассмотрению другого кластера.

Методы ***поиска по ключевым словам*** подразделяются на следующие группы:

1. последовательный поиск;
2. точный поиск по алгоритму;
3. интеллектуальные методы поиска.

Поиск по ключевым словам (терминам) легко реализуем и эффективен, поэтому он получил наибольшее распространение.

Методы последовательного поиска применяют ***методы сканирования словаря*** с использованием нечёткого поиска или формирование ***поискового индекса*** на основе исходной информации, логической и физической структуры данных.

Методы, использующие поисковый индекс можно разделить на 3 группы:

1) хеширующие методы;
2) инвертированные файлы (ИФ) [6,7,8,9]
3) сигнатурные файлы (СФ) [7,8, 9].

Хеширующие методы поиска основаны на идее, которая заключается в следующем: пусть на множестве ключевых терминов документов задана функция $H(w_i)$, отображающая множество исходных терминов документов в конечный отрезок целых чисел $[n...m]$, и пусть $n \le m$. Выделяют множество ячеек памяти для $(m - n + 1)$ терминов и для каждого термина w_i помещают его в ячейку под номером $H(w_i)$. Если $H(w_i) \ne H(w_j)$ для всех $i \ne j$, то для поиска произвольного термина w_i потребовалось бы только одно обращение к элементу массива под номером $H(w_i)$. Но поскольку построение хеш-функции на множестве терминов довольно сложная задача, на практике допускаются коллизии хеширования, когда $H(w_i) = H(w_j)$ для некоторых пар $i \ne j$. Простота реализации и высокая скорость поиска на достаточно небольших объёмах данных являются основными преимуществами хеширующих методов поиска.

Инвертированные файлы логически устроены подобно указателю в конце книги. Они представляют собой список ключевых терминов, где каждому термину сопоставлен список его вхождений в документы. Каждое вхождение ключевого термина определяется ссылкой на исходный документ, содержащий данное ключевое слово. Инвертирование является полным, если в списке вхождений терминов помимо ссылок на документы содержится информация о точном местоположении каждого термина, однако в этом случае размер индекса может быть очень велик и в несколько раз превышать суммарный размер исходных документов. Поэтому в большинстве систем термины в инвертированном списке ссылаются на документы как таковые и не указывают на точное местоположение термина в документе. Поиск происходит посредством сканирования инвертированного списка ключевых слов для нахождения соответствующих терминов запроса и документов и последующего объединения или пересечения результатов, полученных для каждого термина поискового запроса. Для уменьшения размеров инвертированного индекса обычно применяют неполное инвертирование, когда вместо ссылок на точный адрес термина хранятся только ссылки на документы или на блоки данных некоторого фиксированного размера, в которых содержится данный термин. Так детализация адреса термина может быть различной: вхождение слова может быть задано на уровне документа, абзаца или даже на уровне точного местоположения в тексте. В современных поисковых системах, основанных на ИФ, применяются также методы сжатия инвертированного индекса, и в настоящее время размер индекса сжатых инвертированных файлов может составлять лишь 4-10% от общего объёма исходных документов. Для ускорения поиска инвертированный словарь ключевых слов в конец документов может быть представлен в виде дерева или хеш-таблицы. Надо сказать, что ИФ не позволяют производить поиск по произвольной подстроке. Также в большинстве индексов, основанных на ИФ, с целью экономии дискового пространства применяют неполное инвертирование. Однако неполное инвертирование ведёт к увеличению вычислительных затрат, например, на поиск по фразам, где необходимо определять точное положение ключевых слов в документах. Более того, довольно проблематично с помощью ИФ индексировать текст, например на японском или китайском языке, поскольку в данных языках не существует чёткого определения слова.

Сигнатурные файлы были разработаны в качестве альтернативы инвертированным файлам, поскольку долгое время не удавалось обеспечить компактность их индекса.

Рассмотрим структуру сигнатурных файлов более подробно. Пусть существует хеш- функция $f(w)$, отображающая множество ключевых слов во множество целых чисел, допустим от 1 до n . Далее каждому документу ставится в соответствие битовый вектор, где i компонент вектора равен 1 тогда и только тогда, когда в документе существует ключевое слово w_k , такое что $f(w_k) = i$. Такой вектор называют сигнатурой документа. Сигнатурный файл является списком сигнатур всех актуальных документов ИПС. Процесс поиска происходит следующим образом: для каждого слова w в поисковом запросе выбираются только те сигнатурные векторы документов, в которых единица стоит в i позиции, где $f(w) = i$, поскольку очевидно, что документы только с такими сигнатурами могут содержать ключевое слово, и далее только эти документы сканируются на наличие ключевого слова.

Основным недостатком данного метода является сканирование при поиске списка всех сигнатур документов. Для преодоления этого ограничения используется метод битовых срезок. Суть метода битовых срезок заключается в построении сигнатуры, позволяющей делать обратное отображение терминов в документы, их содержащие. В данном случае строится вектор длины n, где n определяется верхней границей определённой выше хеш-функции, и для каждого элемента i данного вектора создаётся битовый вектор длины M , где M – количество исходных документов, в котором единицы соответствуют номерам тех документов, которые содержат хотя бы одно слово w, удовлетворяющее условию $f(w) = i$. Поиск по битовым срезкам происходит следующим образом: для всех ключевых слов запроса вычисляются значения хеш-функции $f(w)$, и далее по этим значениям

выбираются соответствующие битовые срезки, которые соединяются между собой с помощью конъюнкции или дизъюнкции (в зависимости от того, в каком отношении находятся термины поискового запроса, а они могут соединяться через OR или AND) и выбирается множество документов-кандидатов, возможно удовлетворяющих запросу. Основное влияние на точность такого рода поиска определяет вид хеш-функции f (w). Ясно, что при большом количестве документов размер битовой срезки может быть очень велик, поэтому применяют метод блочной битовой срезки, в которой i элемент срезки соответствует не одному документу, а СФ разрабатывались как альтернативна по отношению к ИФ с более компактным индексом. Размер СФ составляет около 50% от размера исходных данных. В настоящее время разработаны методы кодирования, позволяющие строить компактные ИФ, поэтому СФ применяются не очень часто.

Методы, использующие сканирование словаря*,** применяют суффиксные массивы и деревья. ***Суффиксные массивы. Хотя инвертированные файлы в настоящее время очень широко используются, они не позволяют производить поиск по произвольной подстроке - поиск в ИФ может производиться лишь с точностью до некоторого ключевого слова. Кроме того, в большинстве индексов, основанных на ИФ, с целью экономии дискового пространства применяют неполное инвертирование, которое ведёт к увеличению вычислительных затрат. Также довольно проблематично с помощью ИФ индексировать текст, например на японском или китайском языке, поскольку в данных языках не существует чёткого определения слова. По той же причине ИФ также не могут осуществлять поиск в последовательностях ДНК. Для решения данных проблем была разработана структура данных, позволяющая производить более детализованные запросы, и также не требующая деления исходных данных на слова или словоформы [10]. В одних источниках эта структура данных фигурирует как суффиксный массив (suffix array), в других как PAT массив (PAT array).

Деревья. Полнотекстовые индексы, базирующиеся на структурах данных, основанных на списках, не могут обеспечить время поиска по текстовым данным лучше, чем $O(\log n)$. К тому же, списки ключевых терминов, позволяющие осуществлять поиск с логарифмическим временем, должны быть отсортированы, т.к. только тогда по ним можно осуществлять бинарный поиск за логарифмическое время. В свою очередь отсортированные списки трудно модифицировать, вставлять и удалять ключевые термины, и количество операций, необходимых для этого, обычно не меньше, чем $O(n)$.

Существуют структуры данных, свободные от упомянутых выше недостатков. Ими являются деревья. Бинарные деревья, например, могут применяться в качестве структуры данных для инвертированных файлов. Для этого необходимы бинарные лексикографические отношения между ключевыми терминами инвертированного списка, т.е. два любых термина w_i и w_j должны быть лексикографически сравнимы для того, чтобы построить бинарное дерево терминов. В этом случае время поиска по шаблону будет логарифмическим и ограничено величиной $O(m \cdot \log n)$. Существует масса примеров применения древовидных структур в задачах полнотекстового поиска (это сравнимо с
вышеуказанным).

Точный поиск по алгоритму Бойера-Мура. Известно несколько способов улучшения описанного выше простейшего решения последовательного поиска. Наиболее известны два из них: алгоритм Бойера-Мура [12] и алгоритм Кнута-Морриса-Пратта [13]. В большинстве случаев при решении задач точного поиска методом последовательного сканирования алгоритм Бойера-Мура работает быстрее и широко применяется в программах редактирования текста. Сравнение терминов в соответствии с идеей алгоритма происходит справа налево и начинается между последним символом шаблона P и m -им символом текста T , где $|P| = m$ - длина шаблона. Далее, если обнаруживается несовпадение на некоторой паре терминов p_i и t_j , соответственно шаблона и текста, тогда определяется, в каких позициях шаблона содержится несовпадающий символ t_j , и на основе этого принимается решение о сдвиге шаблона относительно текста. Реализация данного алгоритма требует предобработки поискового шаблона перед началом поиска за время $O(m)$. И сам поиск происходит за время $O(m \cdot n)$, но в лучшем случае поиск алгоритмом Бойера-Мура может производиться за время $O(n/m)$. Полное описание работы данного алгоритма можно найти в оригинальной публикации [12] .

Интеллектуальные методы поиска. Основной задачей, возникающей при работе с полнотекстовыми документами, является поиск документов по их содержанию. Однако ставшие традиционными средства контекстного поиска по вхождению слов в документ, представленные, в

частности, поисковыми машинами в Интернет, зачастую не обеспечивают адекватного выбора информации по запросу пользователя. Основная проблема заключается в сложности точной формулировки запроса – подбора ключевых слов, которые предстоит искать в телах документов. Это может быть связано с рядом причин: недостаточным знанием пользователем терминологии предметной области, наличием в языке многозначных и синонимичных слов и даже орфографическими ошибками в написании искомых слов, которые могут встречаться как в текстах, так и в самом запросе.

Существуют различные методы структуризации текста, такие как: гипертекст, семантические сети, методы массированной онтологии концептуальных значений, частотно-вероятностные и логико-статистические модели и методы, метод рубрицирования, метод автоматического нелингвистического анализа неструктурированной текстовой информации, реализованный на основе нейросетевых алгоритмов и т.д. [14].

Так как при обращении к поисковой системе пользователь должен иметь возможность получить в ответ ресурсы, релевантные смыслу запроса, то их поиск должен быть семантически ориентированным. Для этого средства поиска соответствующей запросу информации предлагается в работе [15] организовать на основе онтологии, содержащей описания семантики ресурсов.

Онтологии – эффективный вариант накопления и представления знаний, являющийся, с одной стороны, достаточно выразительным, с другой стороны, естественным и легким для понимания. Онтология определяется как иерархия классов и объектов плюс логическое описание свойств и механизмов взаимодействия этих объектов. Формализация онтологий и “встроенная” логика позволяют работать с ними не только человеку, но и компьютеру.

Известно, что семантика информационных ресурсов очень разнообразна, следовательно, осуществлять поиск необходимой информации тем проще, чем уже и специфичнее предметная область. Вследствие этого на практике можно ограничиться построением онтологии одной конкретной области.

Для построения онтологии требуется формальное декларативное представление четко организованных конструкций, которые включают в себя словарь терминов тематической области, описание определений этих терминов, существующие взаимосвязи между ними, их теоретически возможные и невозможные взаимосвязи. Описанные таким образом онтологии предлагается применить в качестве посредника между пользователем и поисковой системой (рис. 2).

Взаимодействие с онтологией предполагается на следующих этапах:

1) формирование поискового образа релевантного документа;
2) построение запроса к поисковой системе;
3) формирование списка релевантных документов.

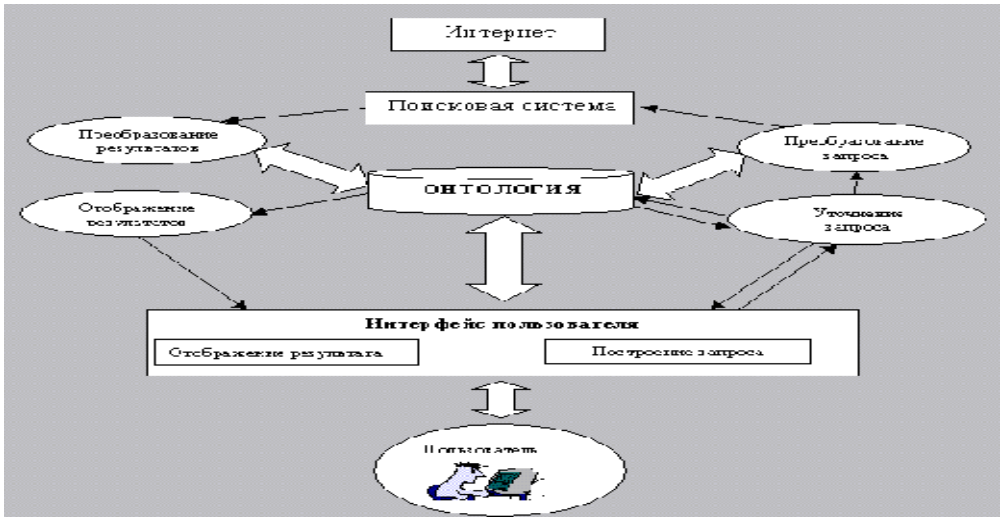

Рис. 2. Схема организации поиска на основе онтологии

Проблема состоит в том, чтобы сделать поиск динамичным и удобным для пользователя. Для любого типа запроса, возникающего у человека в практической деятельности, должны быть найдены адекватные знания в информационном пространстве. При этом язык для формулирования поискового требования не должен был слишком сложным. В частности, общение пользователя с поисковой системой можно сделать более простым, приблизив язык запроса к естественному языку.

При такой организации поиска на этапе формирования образа релевантного документа из пользовательского запроса выделяются смысловые структуры: значимые слова и термины предметной области. Эти смысловые структуры затем используются для формирования поискового образа с применением эвристических правил и вывода на онтологии.

Образ релевантного документа представляет собой описание желаемого результата работы поисковой системы, которое включает в себя:

1) набор терминов, которые должны включаться в документ;
2) набор характеристик документа;
3) набор требований к результату поисковой системы, таких как количество документов и т.п.

На этапе построения запроса к поисковой системе осуществляется вывод на онтологии. При этом выполняется преобразование пользовательского запроса в соединенный логическими связками набор терминов и понятий, которые будут использоваться поисковой системой [15].

Семантический поиск с использованием онтологии возможен в двух вариантах:

1) поисковый запрос совпадает с названием какой-либо концепции в онтологии;
2) поисковый запрос является подмножеством словаря онтологии.

Обзор существующих методов поиска релевантных документов позволяет сделать вывод о том, что в поисковых системах помимо улучшения реализации задач поиска по текстовым ресурсам, решаются также задачи смыслового поиска информации в различных информационных ресурсах. При этом поиск информации по смыслу рассматривает возможность наряду с построением семантических сетей документов применение онтологических знаний предметных областей.

Литература

1. Адаманский А. Обзор методов и алгоритмов полнотекстового поиска [Электронный ресурс]. - www.dialog-21.rudialog2006materialshtmlFedorovsky.htm.
2. Rijsbergen C.J. Information Retrieval [Электронный ресурс]. - http://www.dcs.gla.ac.uk/Keith/Preface.html.
3. Graham, Stephen String Search [Электронный ресурс]. -http://learn.at/infoscope/string_sea...-92/index.html
4. Navarro G. Approximate Text Searching [Текст]// Technical Report TR/DCC-98-14, 1998
5. Бойцов Л. М. Обзор поисковых методов [Электронный ресурс]. - http://alglib.sources.ru/articles/search.php
6. Озкарахан Э. Машины баз данных и управление базами данных [Текст] - М.: Мир, 1989.
7. Faloutsos C., Oard D. «Survery of Information Retrieval and Filtering Methods»[Электронный ресурс]. - http://en.scientificcommons.org/109061
8. Zobel J., Moffat A., Ramamohanarao K. Inverted files versus signature files for text indexing [Текст] -Collaborative Information Technology Research Institute, Departments of Computer Science, RMIT and The University of Melbourne, Australia, feb 1995, Technical report No TR-95-5
9. Rijsbergen C.J. Information Retrieval [Текст] - London: Butterworths, 1979
10. Manber U., Myers G. Suffix arrays: A new method for on-line string searches [Электронный ресурс]. - http://webglimpse.net/pubs/suffix.pdf
11. Захарова И. В. Метод организации семантического поиска документов в электронных библиотеках [Электронный ресурс] - do.csu.ru/~iren
12. Boyer R.S., Moore J.S. A fast string searching algorithm [Текст] - Communications of the ACM. 20:762-772, 1977.
13. Knuth D.E., Morris J.H., Pratts V.R. Fast pattern matching in strings [Текст] - SIAM J. ,1977 Comput. 6, 322-350.
14. Ярных Ю.А. Структурированная семантическая модель контента текстов научно-теоретического характера [Текст]// Дис.... канд. техн. наук. М., 2005.-116 с.

15.Россеева О.И., Загорулько Ю.А. Организация эффективного поиска на основе онтологий [Электронный ресурс] - www.dialog-21.ru/Archive/2001/volume2/2_49.htm