

Саитов Н.Ж.

**ОПТИМИЗАЦИИ ПОИСКА В РЕЛЯЦИОННЫХ БАЗАХ
ДАНЫХ С ИСПОЛЬЗОВАНИЕМ СЕМАНТИЧЕСКОЙ СЕТИ И ФРЕЙМОВОЙ
МОДЕЛИ ПРЕДСТАВЛЕНИЯ ЗНАНИЙ**

N.Zh. Saitov

**SEARCH ENGINE OPTIMIZATION IN RELATIONAL
DATABASES USING SEMANTIC NETWORK AND FRAME MODEL
OF KNOWLEDGE REPRESENTATION**

УДК.: 004.031.2

Обосновано использование семантико-фреймовой модели представления знаний для оптимизации поиска в реляционных базах данных.

Justified the use of semantic-frame model of knowledge representation for search engine optimization of relational databases.

Введение. В последнее время наблюдается возрастание интереса к искусственному интеллекту, вызванное повышением требований к информационным системам. Умнеет программное обеспечение, умнеет бытовая техника. Мы неуклонно движемся к новой информационной революции, сравнимой по масштабам с развитием Интернета, имя которой – искусственный интеллект.

Рождение искусственного интеллекта как научного направления произошло только после создания ЭВМ в 40-х годах XX века. В то же время Норберт Винер создал свои основополагающие работы по новой науке – кибернетика. Термин искусственный интеллект (artificial intelligence) предложен в 1956 г. на семинаре с аналогичным названием в Стэндфордском университете (США). Семинар был посвящен разработке логических, а не вычислительных задач.

В настоящее время различают два основных подхода к моделированию искусственного интеллекта (AI – artificial intelligence): машинный интеллект, заключающийся в строгом задании результата функционирования, и искусственный разум, направленный на моделирование внутренней структуры системы. Основная задача интеллектуальных технологий сегодня – это обработка знаний. Чаще всего интеллектуальные системы применяются для решения сложных задач, где основная сложность решения связана с использованием слабо формализованных знаний специалистов-практиков и где логическая (или смысловая) обработка информации превалирует над вычислительной. Например, понимание естественного языка, поддержка принятия решения в сложных ситуациях, постановка диагноза и рекомендации по методам лечения, анализ визуальной информации, управление диспетчерскими пультами и др.

Разработка интеллектуальных программ суще-

ственно отличается от обычного программирования и ведется путем построения системы искусственного интеллекта (СИИ). Если обычная программа может быть представлена в парадигме:

Программа = Алгоритм + Данные,

то для СИИ характерна другая парадигма:

СИИ = Знания + Стратегия обработки знаний.

При изучении интеллектуальных систем традиционно возникает вопрос – что же такое знания и чем они отличаются от обычных данных, десятилетиями обрабатываемых ЭВМ. Можно предложить несколько рабочих определений, в рамках которых это становится очевидным.

Данные – это отдельные факты, характеризующие объекты, процессы и явления предметной области, а также их свойства. *Знания* основаны на данных, полученных эмпирическим путем. Они представляют собой результат мыслительной деятельности человека, направленной на обобщение его опыта, полученного в результате практической деятельности.

Знания – это закономерности предметной области (принципы, связи, законы), полученные в результате практической деятельности и профессионального опыта, позволяющие специалистам ставить и решать задачи в этой области.

Часто используется такое определение знаний.

Знания – это хорошо структурированные данные, или данные о данных, или метаданные. В отличие от данных знания обладают следующими свойствами:

- *внутренней интерпретируемостью* - вместе с информацией в базе знаний представлены информационные структуры, позволяющие не только хранить знания, но и использовать их;
- *структурированностью* – выполняется декомпозиция сложных объектов на более простые и установление связей между ними;
- *связанностью* – отражаются закономерности относительно фактов, процессов, явлений и причинно-следственные отношения между ними;
- *активностью* – знания предполагают целенаправленное использование информации, способность управлять информационными процессами по реше-

нию определенных задач.

Использование традиционных языков программирования при решении задач не позволяло отделить знания от прикладной программы. Однако подобные методы затрудняли понимание, каким образом используются знания, и какую роль они выполняют. Особенность систем представления знаний заключается в том, что они моделируют деятельность человека, осуществляемую часто в неформальном виде. Модели представления знаний имеют дело с информацией, получаемой от экспертов, которая часто носит качественный и противоречивый характер. Для обработки с помощью ЭВМ такая информация должна быть приведена к однозначному формализованному виду. В системах, основанных на концепциях искусственного интеллекта и инженерии знаний, такая проблема отсутствует. Знания в них представлены в конкретной форме, а имеющаяся база знаний позволяет их легко определять, модифицировать, пополнять [1].

Постановка задачи. В данной статье рассматриваются понятия семантической сети и фреймовая модель представления знаний для оптимизация поиска в реляционной базе данных.

Результаты исследований. В целях анализа задачи рассмотрим модели представления знаний. Все модели и системы представления знаний (МПЗ и СПЗ) можно разделить на следующие классы: декларативные, процедуральные, специальные. В каждом классе моделей можно выделить свои подклассы: в декларативных СПЗ - продукционные, редукционные и предикатные модели; в процедуральных – язык PLANNER и ему подобные языки; в специальных – логико-лингвистические модели, модели реляционной алгебры и алгебры нечетких множеств, функциональные семантические сети и т.д. В *декларативных* представлениях описание состояний представляет собой множество утверждений, в значительной степени независимых от того, где их использовать. В *предикатных системах* появляется аспект выводимости (импликативности), т.к. в них используется система доказательства теорем, и они носят, на первый взгляд, черты процедуральных систем. Но т.к. процесс доказательства состоит в применении заданных заранее правил вывода к заданным заранее начальным аксиомам, а затем этих и других правил вывода к аксиомам и выведенным на первом шаге теоремам и т. д., вплоть до получения целевой формулы, то предикатные системы чаще относят к декларативным представлениям. При декларативном представлении имеет место четкое разделение процедуры поиска решения, приводящей к полному перебору (называемой *механизмом генерации*), и процедуры оптимизации этого поиска с целью сокращения перебора (называемой *механизмом управления*). Механизм генерации не зависит от ПО и носит универсальный характер. При этом, как правило, приходится на каждом шаге иметь дело с полными описа-

ниями состояний, что снижает эффективность поиска. Механизм управления определяет эвристическую эффективность поиска вывода. Он, как правило, определяется знаниями о конкретной ПО и определяет семантику декларативной СПЗ. Знаний. Описания состояний и операторов системы носят синтаксический характер. Поэтому можно утверждать, что синтаксические знания в декларативных представлениях отделены от семантических [2].

Все ранее рассмотренные свойства характеризуют декларативные представления как обладающие принципиально большой общностью, универсальностью по отношению к любым ПО. Но для придания эффективности поиска механизм управления требует конкретных семантических знаний о ПО, что автоматически сужает общность такого представления. Если вводить семантические знания в описания состояний и операторов (так, как это делается в других представлениях), то теряется общность и универсальность представления. Здесь наблюдается известное *противоречие между общностью и эффективностью* представления знаний. При *процедуральном* представлении знаний задаются процедуры преобразований знаний для данной ПО. При этом текущее состояние системы представляется в виде набора специализированных процедур, обрабатывающих локальный участок БД, которая специально организована, как говорят, в *императивном* виде, т.е. в приспособленном для использования системой процедур и доказательства теорем (термин императивный используется в лингвистике для обозначения сослагательного наклонения, в котором, как правило, используется частица «бы», например, «я выучил бы язык PLANNER, если бы у меня было время»). Процедуральные представления знаний, как правило, реализуются с помощью специальных языков типа PLANNER, QA-4 и т.п., в которых имеется набор механизмов, обеспечивающих автоматический и целенаправленный поиск решения. В этих языках обычно стараются оптимальным образом разрешить *проблему сочетания универсальности механизмов и управляемости* ими со стороны пользователя. В процедуральных представлениях знаний семантическая информация о конкретной ПО вводится в выражения базы данных в виде их свойств, что позволяет использовать на данной стадии поиска решения только те данные, которые необходимы действующей (активизированной) в данный момент процедуре и в ней описаны. Одним из характерных свойств процедурального представления знаний, которое внешне его сразу отличает от декларативного, является представление информации, в частности, описаний состояний в пространстве состояний в виде процедур, программ. Так, вместо хранения информации о всех возможных состояниях можно хранить программу, которая сгенерирует из начального состояния все возможные. [3]

Такое хранение информации существенно эко-

номит память ЭВМ. Однако, процедуральные представления знаний, обеспечивая в целом более быстрый поиск решения по сравнению с декларативными, обладают меньшими средствами для накопления, усвоения, пополнения знаний, что при наличии в них механизма недетерминированного поиска (выбора) делает их менее приспособленными к решению задач в сложных и не полностью известных ПО. Декларативные представления знаний, обладая большой универсальностью и общностью, в этих случаях являются более предпочтительными. Разделение синтаксических и семантических знаний в моделях представления знаний, четко обозначенное в декларативном и менее в процедуральном представлениях, имеет свои положительные и отрицательные стороны. Для соединения этих двух типов знаний, в результате которого в итоге повышается эффективность системы, было разработано большое количество разных вариантов моделей представления знаний в виде *семантических сетей*. Практически исключая разделение синтаксических (структурных) и семантических (привязанных к данной ПО) знаний, это представление позволяет достаточно легко пополнять, обновлять, усваивать знания в относительно однородной структуре. В этом представлении описание внешнего мира осуществляется на языке, близком естественному. Благодаря общей системе унификации отношений между объектами существенно упрощается процедура вывода решения, при которой автоматически определяются участки семантической сети, необходимые для интерпретации, понимания запроса и вывода решения. Однако, для представления простых логических отношений типа логических связок, кванторов общности и существования семантические сети уступают декларативному представлению знаний, а в части отображения динамических и параллельных процессов - процедуральному. Поэтому стали использовать удачные приёмы, как из декларативных, так и процедуральных представлений знаний, что нашло своё воплощение в *сетях фреймов* (frame - каркас, рамка). Характерной особенностью этого представления знаний является введение в систему представления знаний модульной структуры в виде фреймов, которые представляют собой семантические блоки (единицы) в общем случае процедурально-декларативного типа. Поэтому фреймы представляют собой локальные семантические сети и являются единицами, атомами информации. В общем случае фреймы содержит информационные и процедуральные элементы, обеспечивающие преобразование информации внутри и связь его с другими фреймами. Но самое главное, как в процедуральной,

так и в информационной части (хотя деление достаточно условное, ибо информация может храниться в виде программ) имеются незаполненные участки, пустоты или, как их называют, *слоты* (в переводе с английского - щели). Слоты могут заполняться в процессе активации, функционирования фрейма в соответствии с определёнными условиями или предписаниями, которыми они сопровождаются. Это придаёт свойство адаптивности, настраиваемости как на модульном, атомарном уровне слотов, так и на уровне всей сети фреймов. Наряду с положительными сторонами сети фреймов вносят в систему представления знаний дополнительные трудности, в частности, связанные с зацикливанием фреймов. Модели представления знаний, использующие *реляционную алгебру*, ведут своё начало от реляционного банка данных. В основе моделей, использующих реляционную алгебру, лежит представление информации в виде сети единиц (атомов) информации, в качестве которых фигурируют отношения (таблицы) или домены (столбцы). Здесь широко используется приёмы декларативного представления знаний, в частности, логические связки и кванторы, а также аспекты процедурального представления знаний, в частности, представление информации в виде процедур, программ. В последних модификациях реляционного банка данных стали использоваться однородные структуры типа семантических сетей с фреймоподобными элементами. Характерной чертой представления знаний с использованием *алгебры нечётких множеств* является его направленность на реализацию свойств нечёткости принятия решений. В своих наиболее совершенных вариантах это представление знаний использует все черты ранее рассмотренных моделей.[4]

Выводы. В результате анализа выявлено целесообразность использования семантической сети и фреймовой модели знаний для оптимизации поиска в реляционной базе данных.

Список литературы:

1. Андрейчиков А.В. Интеллектуальные информационные системы: Учебник / Андрейчикова О.Н. // М.: Финансы и статистика, 2004. – 424 с.
2. Гаврилова Т.А. Базы знаний интеллектуальных систем. / Хорошевский В.Ф. // СПб.: Питер, 2001 – 384с.
3. Лапина А. В.- руководитель проекта. Интеллектуальные информационные системы
4. Джарратано, Джозеф. Экспертные системы: принципы разработки и программирование. / Райли, Гарри // 4-е издание. М.: Вильямс, 2005. - 1152 с.

Рецензент: к.ф.-м.н., доцент Кыдыралиев Н.Н.