

где $est1_{t-1}$, $est2_{t-1}$ – отклонения от долгосрочных зависимостей; ler – логарифм номинального рыночного обменного курса; $D9804$, $D9801$, $D9809$, $D0002$ – фиктивные переменные. Коэффициенты корректировки являются значимыми и имеют отрицательные знаки, что соответствует экономическому смыслу. Таким образом, можно сделать вывод о функционировании механизма восстановления долгосрочного равновесия.

Анализ остатков указал на отсутствие автокорреляции: минимальное p -значение для LM-теста равно 0.17, для Portmanteau-теста равно 0.39. Совместная статистика Жака-Бера показала невозможность отклонить гипотезу о нормальном распределении остатков (p -значение равно 0.13). Для статистики Вайта p -значение равнялось 0.57, что свидетельствует об отсутствии гетероскедастичности.

Литература

1. *Banerjee A., Russell B.* A markup model for forecasting inflation in the euro area // European Central Bank. Working paper No 306, 2004.
2. *Johansen S.* Likelihood-Based Inference in Cointegrated Vector Autoregressive Models / Oxford University Press, 1996.
3. *Sriram S. S.* Survey of Literature on Demand for Money // Washington: International Monetary Fund), IMF Working Paper 99/64, 1999.
4. *Пелюпась И.* Спрос на деньги и инфляция в Беларуси // Эковест. 2001. №1. – С. 6–63.
5. *Харин Ю. С., Малюгин В. И., Пранович М. В., Мурын Д. Л.* Система эконометрических моделей для прогнозирования и оценки вариантов денежно-кредитной политики // Белорусский экономический журнал. 2003. №3. – С. 24–45.

РЕФЕРИРОВАНИЕ КАК ЗАДАЧА ИНЖЕНЕРИИ ЗНАНИЙ

Н. В. Воронков

Понятие «управление знаниями» (УЗ) появилось в середине 90-х годов в крупных корпорациях, для которых проблемы обработки информации приобрели особую остроту и стали критическими. Вскоре оказалось очевидным, что основным узким местом здесь является работа (прежде всего, сохранение, поиск, тиражирование) со знаниями, накопленными специалистами компании, так как именно этот ресурс обеспечивает ей преимущество перед конкурентами. Зачастую информации в компании накоплено гораздо больше, чем она способна оперативно переработать. При этом часто одна часть предприятия дублирует работу другой только потому, что невозможно найти и использовать знания, накопленные в соседних подразделениях.

Управление знаниями можно рассматривать как направление в менеджменте (стратегия, обеспечивающая интегрированный подход к созданию, организации, использованию и увеличению интеллектуальных и

информационных ресурсов предприятия), и как направление в информатике (поддержка процессов создания, распространения, обработки и использования знаний) [1].

Безусловно, разработка систем управления знаниями ориентирована на определенный подход и модель (язык) описания данных и знаний. В этом плане, все большую популярность в последнее время получают онтологии. Онтология – это точная спецификация некоторой области, которая включает в себя словарь терминов предметной области и множество связей (типа элемент-класс, часть-целое и т.д.), которые описывают, как эти термины соотносятся между собой. Фактически это иерархический понятийный скелет предметной области.

В настоящее время практически все накопленные знания доступны в виде печатного материала и в виде электронных документов. Поэтому роль автоматической обработки текстов очень высока и продолжает расти. Часто в процессе работы с онтологиями экспертам необходимы ссылки на первоисточники из которых выделены те или иные знания. Как правило, этими источниками опять-таки являются тексты в электронном виде. Однако количество этих ссылок может быть очень большим и время затрачиваемое на их обработку может стать серьезным препятствием для работы с онтологическими системами. Возникает необходимость сократить объем информации, который должен быть обработан экспертом. В связи с этим появляется задача реферирования текстов.

В общем случае под рефератом понимается связное изложение основных идей текста. Реферирование можно рассматривать и в контексте классических элементов, выделяемых системами обработки знаний, т.е. как систему, ориентированную на работу с объектами, фактами и закономерностями предметной области.

Таким образом, системы реферирования должны иметь механизм выделения указанных элементов, а также уметь упорядочивать их в соответствии с уровнем релевантности к документу и выделять фрагменты текста, описывающие эти элементы.

Однако, более эффективным будет построение не отдельных рефератов ориентированных на объекты, факты и закономерности предметной области, а создание единого процесса реферирования, в результате чего можно при построении одних типов рефератов использовать их преимущества в других.

В качестве общей схемы выделения элементов базы знаний из текстового документа можно предложить следующую:

Преформатирование. Существует много различных форматов документов и поэтому для упрощения процесса их обработки возникает необходимость конвертации их в некоторый формат, который должен быть удобен для обработки, а так же максимально сохранять стилистическую

и структурную разметку документов. А также происходит разбиение текста на параграфы, выделение заголовков и подзаголовков, выделение разделов текстов. Также происходит фильтрация вспомогательного текста – текстов кнопок, меню, скриптов и т.д.

Лингвистический (лексический, лексико-грамматический, синтаксический и семантический) анализ текста, в результате которого определяются лексико-грамматические классы его слов, строится синтаксическое дерево каждой фразы, распознаются объекты и семантические отношения между ними типа С-А-О (субъект-акция-объект) и отношения типа Причина-Следствие – между самими САО-тройками [2]. Понятно, что САО-тройка соответствует такому классическому элементу знаний как факт, а Причина-Следствие – закономерности предметной области.

Обработка лексико-статистической информации. На этом этапе происходит накопление статистических весов информативных слов текста (к неинформативным словам относятся такие части речи как предлоги, числительные и некоторые другие). Причем, статистика должна собираться с использованием как отдельных слов, так и объектов, фактов и закономерностей.

Для создания реферата на уровне объектов, удобным и достаточно эффективным является построение так называемого topic-ориентированного реферата, который позволяет выделить наиболее важные объекты (в качестве объектов берутся именные группы) документов, а также построить иерархическое дерево тем и получить “урезанные” предложения описывающие интересующие пользователя темы. Подробно процесс создания topic-ориентированного реферата описан в [3]. В качестве предварительной обработки текста используются шаги описанные выше, а затем, посредством обработки выделенных именных групп (трансформация, фильтрация, канонизация) и назначения им весов в соответствии с определенным алгоритмом, среди всех именных групп выделяются наиболее важные – темы. После этого происходит построение иерархического дерева и выделение главных тем на основе этого дерева. Далее выделяются наиболее информативные предложения, описывающие эти темы и происходит «усечение» этих предложений с учетом тем. Причем для выделения наиболее важных предложений подходит метод, используемый при построении так называемого классического реферата [4]. В основе его построения лежит комбинация лингво-статистического, позиционного и эвристического алгоритмов, в результате чего происходит комплексная оценка весов предложений для вычисления их релевантности к тексту. Лингво-статистический алгоритм представляет собой построение реферата, основанного на фактах и закономерностях, выделенных из предложений текста посредством оценки их статистической важности. Поскольку закономерности по сути представляют из себя 2

факта – причину и следствие, то удобным является ограничиться вычислением статистической оценки фактов, входящих в текст, а затем провести оценку закономерностей уже на основе имеющейся оценки фактов. Позиционный метод позволяет учесть местоположения предложений в тексте, а также учесть расположение фактов и закономерностей внутри предложений. Из-за субъективизма человеческих оценок, разные эксперты часто не могут дать одинаковой оценки предложениям. Также не всегда эксперты могут прийти к согласию какая из частей предложения является определяющей с точки зрения ее смысла. В связи с этим и возникает необходимость использования неких эвристических оценок при выделении информативных предложений. В качестве алгоритма дающего эвристическую оценку целесообразно использовать алгоритм слов-подсказок, имеющий гибкий язык правил, позволяющий использовать синтаксические и семантические отношения выделенные из текста, различные словари, элементы документа, не находящиеся непосредственно в рассматриваемом предложении, раздел документа, в котором находится предложение, бонусную систему и систему штрафов. Однако из-за специфики данного метода особое внимание следует уделить методике создания слов-подсказок и начисления им весов [4].

В результате, все предложения получают комплексную оценку, что дает возможность выбора наиболее релевантных предложений описывающих как объекты, так и факты и закономерности.

Таким образом задача реферирования как задача инженерии знаний должна иметь механизм выделения основных составляющих базы знаний: объектов, фактов и закономерностей предметной области (внешнего мира) и включать построение реферата ориентированного на эти 3 основных элемента. Единый механизм построения этих типов реферата позволяет использовать преимущества построения одних типов реферата в других и получить единообразную систему выделения наиболее релевантных предложений, описывающих элементы базы знаний, а также сделать рефераты, ориентированные на различные элементы базы знаний взаимосвязанными.

Литература

1. *Гаврилова Т.А.* Онтологический подход к управлению знаниями при разработке корпоративных информационных систем // *Новости искусственного интеллекта.* 2003. №2. С. 24–30.
2. *Batchilo L.S., Sovpel I.V., Tsourikov V.M.* Document semantic analysis/selection with knowledge creativity capability utilizing subject-action-object (SAO) structures / US Patent 6,167,370, December, 2000.
3. *Voronkov N.V., Sovpel I.V.* Automatic topic-oriented summarization // *Proc. of the Text Processing and Cognitive Technologies Symposium. Paper Collection №7.* Kazan, 2002. P. 94–102.

4. *Воронков Н.В.* Использование эвристических оценок в задаче автоматического реферирования текстов // Информационные системы и технологии: Тез. докл. 1-ой междунар. конф. Мн., 2002.

ВОПРОСЫ РЕАЛИЗАЦИИ КОМПОНЕНТ БИЗНЕС-ЛОГИКИ

А. Л. Дашкевич

В данной статье рассматриваются вопросы применения наиболее перспективных информационных технологий и средств позволяющих автоматизировать процесс разработки WEB-приложений использующих платформу J2EE. Анализируются возможности применения аспектно-ориентированного подхода, для новых возможностей повторного использования кода. Приводится пример применения аспектного подхода для повторного использования компонента, реализующего шаблон Audit Log и его сравнение с другими реализациями.

Большинству разработчиков хорошо известна концепция «слоев» разделения сложного приложения на слабо связанные между собой модули [1, с. 47]. Такое разбиение позволяет в будущем более легкий переход при замене технологии в одном из «слоев» системы, делает систему гибкой к использованию новых технологий и средств. При разработке WEB приложений хорошо известна схема разбиения на «слои» представленная на рис. 1. На этом же рисунке для каждого из «слоев» описаны технологии, предоставляющие готовые каркасы, которые разработчик лишь должен наполнить необходимой логикой.

Нibernate представляет собой перспективный некоммерческий проект, который предоставляет каркас для отображения объектов в реляционную базу данных и язык манипулирования объектами транслирующийся в SQL [2].

Центральным компонентом типичного web приложения является бизнес уровень. Существует многообещающая технология Spring с открытым исходным кодом, которая позволяет определить, каким образом необходимо связывать бизнес объекты с остальными уровнями приложения [3]. Основной подход, который используется в Spring, это «Инверсия по управлению» (Inversion of Control), что делает его одним из наиболее гибких по управлению.

Spring имеет в своем ядре мощную поддержку АОП. Это позволяет по-новому посмотреть на повторное использование кода. Например, существует компонент (acegi-security), реализующий сквозную функциональность для аутентификации и авторизации пользователей в системе, базирующейся на Spring.