УДК 330.43

Informatics and Mathematical Methods in Simulation
Vol. 2 (2012), No. 1, pp. 77-86

РАЗРАБОТКА СЕМАНТИЧЕСКОГО ЯДРА САЙТА С ДИНАМИЧЕСКИМ КОНТЕНТОМ НА ОСНОВЕ АССОЦИАТИВНЫХ ПРАВИЛ

Е.А. Арсирий, О.А. Игнатенко, А.А. Леус

Одесский национальный политехнический университет,
просп. Шевченко, 1, Одесса, 65044, Украина; e-mail: o-ignatenko@mail.ru

В результате анализа проблем продвижения в поисковых системах веб-ресурсов с динамическим контентом предложена методика разработки семантического ядра сайта на основе создания ассоциативных правил с помощью алгоритма поиска популярных наборов Apriori в транзакционной базе данных поисковых запросов. Применение методики позволило повысить полноту и точность, а также снизить время разработки семантического ядра сайта типа Интернет-витрины и магазина.

Ключевые слова: поисковая система, поисковые запросы, семантическое ядро сайта, популярные наборы, ассоциативные правила, алгоритм Apriori

Введение

В век информационных технологий успех практически любого бизнеса в достаточно большой степени зависит от способов виртуального представления фирмы в сети Интернет. При этом целью разработки контента веб-ресурса фирмы является предоставление информации, которая была бы способна заставить пользователя думать и вести себя в направлении, выгодном реальному бизнесу. С другой стороны, известно, что доля «поискового трафика» любого сайта (число посетителей, пришедших от поисковых выдач, от общей посещаемости сайта) является преобладающей [1,2]. Поэтому при разработке контента сайта большое внимание уделяется SEO (*search engine optimization*) – комплексу мер, направленных на продвижение веб-ресурса к верхним позициям поисковой системы (ПС) с целью увеличения его посещаемости. Известно, что одним из ключевых этапов SEO является разработка семантического ядра сайта (СЯС), которая, как правило, выполняется специалистами вручную и требует больших временных затрат [3, 4]. Такое положение является особенно недопустимым при разработке СЯС с динамическим контентом, когда SEO-специалисты не успевают вовремя реагировать на изменяющиеся наполнение сайта, внешнее Интернет-окружение, а также предпочтения и действия пользователей. Поэтому актуальным является создание методики разработки СЯС, применение которой SEO-специалистами позволило бы сократить время на достижение и поддержание лидирующих позиций сайта в поисковых выдачах, что является целью настоящей работы. Для разработки методики авторам необходимо было решить ряд задач:

- проанализировать опосредованную связь между этапами и процедурами работы ПС и разработкой СЯС и предложить метод ее описания;
- определить требования к формированию транзакционной базы данных в терминах анализа связей и разработать базу данных для поисковых транзакций;
- разработать методику применения анализа связей в транзакционной базе поисковых запросов;

- предложить методику реализации поиска популярных наборов с помощью алгоритма *Apriori* и разработки АП на основе найденных популярных наборов для разработки СЯС.

Анализ этапов и процедур работы поисковых систем и разработки семантического ядра сайта

ПС представляет собой сайт, состоящий из веб-интерфейса для пользователя и поисковой машины, которая является движком, обеспечивающим функциональность ПС. Поисковая машина состоит из модуля индексирования, базы данных (БД) проиндексированных документов и поискового сервера, занимающегося анализом и обработкой запросов пользователей. Модуль индексирования состоит из трех вспомогательных программ (роботов) – *spider* (паук), *crawler* (путешествующий паук) и *indexer* (индексатор). *Spider* скачивает веб-документы с помощью протокола HTTP, извлекает ссылки и перенаправления и сохраняет текст в следующем формате: URL, дата скачивания, http-заголовок ответа сервера, тело страницы (html-код). *Crawler* обрабатывает найденные пауком ссылки и осуществляет дальнейшее направление паука. *Indexer* разбирает html-код страницы на составные части такие как заголовки (*title*), подзаголовки (*subtitles*), метатэги (*meta tags*), текст, ссылки, структурные и стилевые особенности и т.д., анализирует их на основе различных лексических и морфологических алгоритмов с целью последующего ранжирования по степени важности. При этом найденным словам и словосочетаниям присваиваются весовые коэффициенты в зависимости от того, сколько раз и где они встречаются (в заголовке страницы, в начале или в конце страницы, в ссылке, в метатэге и т.п.). В результате формируется файл, содержащий индекс, который может быть довольно большим. Для уменьшения его размеров прибегают к минимизации объема информации и сжатию файла, а также решают задачи определения дубликатов и «почти дубликатов». Результаты индексирования записываются в базу данных (БД) проиндексированных документов (рис. 1, а).

Поисковый сервер является важнейшим элементом всей ПС, так как от алгоритмов, которые лежат в основе его функционирования, зависит качество и скорость поиска. Принцип его работы заключается в следующем. Полученный от пользователя запрос (ключевые слова) подвергается морфологическому анализу для получения информационного окружения. При этом выделяются информационные (поиск сведений), транзакционные (совершение действия), нечеткие (общие) и навигационные (прямой адрес) запросы. Поиск документов по их содержанию называется семантическими. Информационное окружение передается специальному модулю ранжирования, задача которого состоит в поиске html-страниц в БД проиндексированных документов, сортировке и выдаче в порядке релевантности. При этом для оценки релевантности найденных документов, как правило, используют *TF-IDF*-меру, согласно которой релевантность документа будет выше, если слово или словосочетание из запроса *чаще* встречается в найденном документе (TF) и *реже* в других документах БД (IDF). Если необходимо, порядок выдачи документов может быть изменен пользователем путем задания дополнительных условий (расширенный поиск). Далее генерируется сниппет, то есть, для каждого найденного документа из таблицы документов извлекаются заголовок, краткая аннотация, наиболее соответствующая запросу и ссылка на сам документ, причем найденные слова подсвечиваются. Полученные результаты поиска передаются пользователю в виде *SERP* (*Search Engine Result Page*) – страницы выдачи поисковых результатов. Таким образом, основой работы всех ПС является определение так называемых «ключевых слов» веб-ресурса. Из списка таких слов состоит семантическое ядро сайта (СЯС). СЯС представляет собой список ключевых слов и их комбинаций, записанных в метатэги *keywords* и распределенных в контенте сайта, а именно, в тэге *title*, в alt-аттрибутах, в ссылочном тексте внутренних и внешних ссылок, в

выделениях жирным и наклонным шрифтом, в начале контента сайта, в названии файлов, в URL и др. При этом от полноты и точности разработки СЯС зависит положение сайта в списке выдач ПС.

Разработка СЯС является ключевым этапом SEO и состоит из ряда интеллектуальных, трудноформализуемых этапов и процедур, для реализации которых необходимы большие временные и человеческие ресурсы (рис. 1, б).

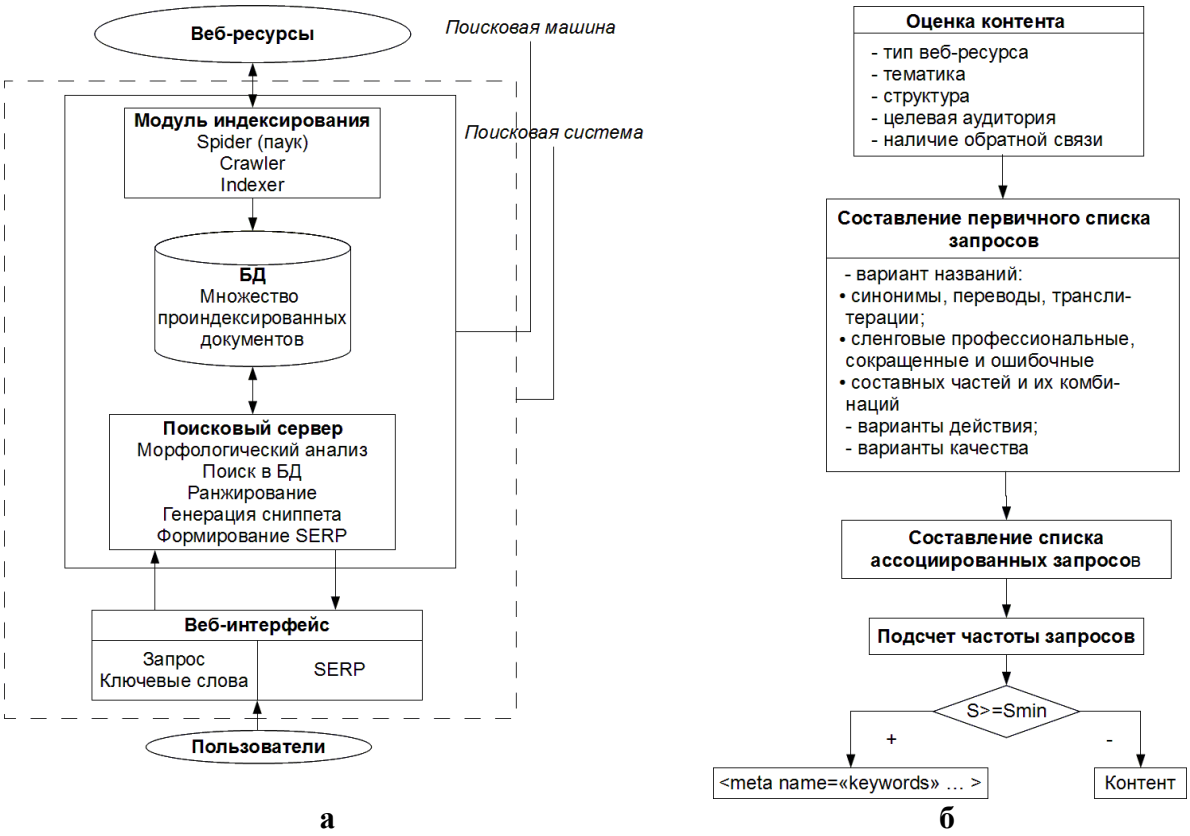

Рис. 1. Обобщенная схема этапов и процедур: а – работы ПС; б – разработки СЯС

На первом этапе необходимо оценить контент сайта, определив его тип (магазин, новостной блог, сайт-визитка и пр.), тематику, структуру, целевую аудиторию и необходимость обратной связи с пользователями. Следующим этапом будет создание первичного списка запросов. Для этого можно использовать различные варианты названий товаров, услуг, самого сайта, различные действия, предоставляемые пользователям и варианты качества товара или услуг [1]. Затем составляется список ассоциированных запросов с помощью средств статистики поисковых систем (wordstat.yandex, adstat.rambler, adwords.google и др.) и подсчитывается частота ключевых слов. Ключевые слова с наибольшей частотой помещают в метатэги *keywords*, с меньшей – распределяют по контенту сайта. Однако, для сайтов с динамическим контентом, таких как Интернет-витрина и магазин, новостной блог, где меняется ассортимент товаров, их популярность, новости, заголовки и пр., перечисленные этапы разработки СЯС необходимо повторять достаточно часто. При этом длительность выполнения каждого этапа может значительно задерживать необходимую периодичность повторения, что приводит к снижению полноты и точности СЯС, а сайт теряет свои позиции в SERP. Для сокращения времени разработки СЯС с динамическим контентом без потери полноты и точности в данном исследовании предлагается использовать анализ связей (*link analysis*), позволяющий сгенерировать правила количественного описания взаимной связи между двумя и более ключевыми словами, объединенными в одном семанти-

ческом запросе. Такие правила в терминах анализа связей называются *ассоциативными*, а запрос представляет собой некоторое множество событий происходящих совместно и образует *транзакцию*.

Транзакционная или операционная БД представляет собой двумерную таблицу, которая состоит из номера транзакции (TID) и перечня ключевых слов, составивших запрос во время этой транзакции. Пример фрагмента транзакционной БД (ТБД) для Интернет-витрины Konica-Digital показан в табл. 1, где TID – уникальный идентификатор, определяющий каждый поисковый запрос или транзакцию. На основе имеющейся транзакционной базы данных необходимо найти закономерности между событиями, то есть запросами пользователей.

Таблица 1.

Транзакционная база данных (фрагмент)

TID	Поисковые запросы			
1	флешки	онлайн		
2	*фото*	*альбом*		
3	*фото*	*альбом*	онлайн	
4	фото	рамки		
5	фото	рамки	купить	онлайн
6	печать	фото	онлайн	
7	фото	магазин		
8	фото	рамки	онлайн	
9	печать	фото		
10	интернет	магазин	фото	
11	*фото*	магазин	*альбом*	
12	фото	магазин	рамки	
13	рамки	*альбом*	*фото*	
14	печать	флешки		

Пусть $I = \{i_1, i_2, i_3, \ldots, i_n\}$ – множество (набор) ключевых слов, называемых элементами. Пусть D – множество транзакций из ТБД, где каждая транзакция T с уникальным номером TID – это набор элементов из I, $T \subseteq I$. Каждая транзакция представляет собой бинарный вектор, где $t[k] = 1$, если i_k элемент присутствует в транзакции, иначе $t[k] = 0$. При этом транзакция T содержит A, некоторый набор элементов из I, если $A \subseteq T$ (табл. 2). Ассоциативным правилом (АП) состоящим из двух наборов элементов называется импликация $A \to B$, где $A \subset I$, $B \subset I$ и $A \cap B = \varnothing$. При этом A называют условием (*antecedent*), а B – следствием (*consequent*) и говорят «если A, то B». Можно выделить объективные (независимые от конкретного приложения) и субъективные (связанные с контекстом задачи) меры значимости АП. К объективным мерам, описывающим связь между наборами элементов, которые соответствуют условию и следствию, относят поддержку – *supp* (*support*) и достоверность *conf* – (*confidence*).

Правило $A \to B$ имеет поддержку *supp*, если *supp%* транзакций из D, содержат $A \cup B$ (условие и следствие), т.е.

$$supp(A \to B) = supp(A \cup B). \quad (1)$$

Достоверность *conf* правила – отношение количества транзакций, содержащих условие A и следствие B, к количеству транзакций, содержащих только условие A – показывает какова вероятность того, что из A следует B

$$conf(A \to B) = supp(A \cup B)/supp(A). \quad (2)$$

При этом говорят, правило $A \to B$ справедливо с достоверностью *conf*, если *conf%* транзакций из D, содержащих A, также содержат B.

Целью анализа связей является получить возможные АП вида $A \to B$ для всех элементов с различными значениями поддержки и достоверности, которые должны быть выше определенных порогов, называемых соответственно минимальной поддержкой (*minsupport*) и минимальной достоверностью (*minconfidence*). При этом следует учитывать случаи, когда условие и следствие являются независимыми ($\text{supp}(A \to B) \approx \text{supp}(A) \cdot \text{supp}(B)$) не смотря на высокие значения поддержки и достоверности. Такое АП не является значимым. Поэтому для оценки значимости правила используют также субъективные меры, которым относят *lift* («интерес») и *leverage* («плечо»).

Lift – отношение частоты появления условия в транзакциях, которые содержат также и следствие, к частоте появления следствия в целом

$$lift(A \to B) = conf(A \to B)/supp(B) = (supp(A \cup B)/supp(A))/supp(B),$$
$$lift(A \to B) = supp(A \cup B)/supp(B) \cdot supp(A). \quad (3)$$

Таблица 2.

Транзакционная база данных 1-элементных наборов в нормализованном виде

Элементы	i_1	i_2	i_3	i_4	i_5	i_6	i_7	i_8	i_9
Значения	флешки	онлайн	фото	альбом	рамки	купить	печать	магазин	интернет
1	1	1	0	0	0	0	0	0	0
2	0	0	1	1	0	0	0	0	0
3	0	1	1	1	0	0	0	0	0
4	0	0	1	0	1	0	0	0	0
5	0	1	1	0	1	1	0	0	0
6	0	1	1	0	0	0	1	0	0
7	0	0	1	0	0	0	0	1	0
8	0	1	1	0	1	0	0	0	0
9	0	0	1	0	0	0	1	0	0
10	0	0	1	0	0	0	0	1	1
11	0	0	1	1	0	0	0	1	0
12	0	0	1	0	1	0	0	1	0
13	0	0	1	1	1	0	0	0	0
14	1	0	0	0	0	0	1	0	0
supp (i_k)	14.3	35.7	85.7	28.6	35.7	7.14	21.43	28.6	7.14
minsupp 28.6%		*itemset*$_1$	*itemset*$_1$	*itemset*$_1$	*itemset*$_1$			*itemset*$_1$	

«Лифт» является обобщенной мерой связи двух наборов элементов. Если $lift(A \to B) > 1$, то связь является положительной, если $lift(A \to B) = 1$ – наборы A и B независимы, если $lift(A \to B) < 1$ – связь является отрицательной.

«Плечо» – разность между наблюдаемой частотой, которой условие и следствие появляются совместно (поддержкой ассоциации) и произведением частот появления условия и следствия в отдельности

$$lever(A \to B) = supp(A \cup B) - supp(A) \cdot supp(B). \quad (4)$$

Реализация анализа связей в ТБД, как правило, включает два этапа:

1) Поиск всех наборов элементов, поддержка которых больше либо равна *minsupp*. Такие наборы элементов называются популярными наборами (*frequent itemset*).

2) Разработка АП на основе популярных наборов с достоверностью большей либо равной *minconf*.

Поиск популярных наборов на основе перебора всех возможных наборов элементов из ТБД потребует $O\left(2^{|I|}\right)$ операций, где $|I|$ – количество элементов, при этом с ростом числа элементов в $I(|I|)$ экспоненциально растет число потенциальных наборов элементов.

Для снижения размерности пространства поиска используют свойство антимонотонности поддержки наборов элементов, заключающееся в том, что поддержка любого набора элементов не может превышать минимальной поддержки любого из его поднаборов. Если все возможные наборы элементов из I можно представить в виде решетки связей, начинающейся с пустого набора, затем на 1 уровне располагаются 1-элементные наборы, на 2-м – 2-элементные и т.д. На k-м уровне представлены k-элементные наборы, связанные со всеми своими (k-*1*)-элементными поднаборами (рис. 2). Предположим, что набор из элементов $\{i_1 i_2\}$ согласно (1) имеет поддержку ниже заданного порога и, соответственно, не является популярным. Тогда, согласно свойству антимонотонности, все его постнаборы также не являются популярными и отбрасываются. На рис. 2 – это часть решетки, начиная с $\{i_1 i_2\}$. Таким образом, любой k-элементный набор будет популярным тогда и только тогда, когда все его (k-*1*)-элементные поднаборы будут популярными.

Разработка АП на основе найденных популярных наборов выполняется после расчета поддержки и достоверности, используя (1) и (2) для всех импликаций типа $A \to B$. При этом в качестве A используются все возможные популярные и непустые (k-*1*)-элементные поднаборы $itemset_{k\text{-}1}$ популярного k-элементного набора $itemset_k$, а в качестве B – разности R между $itemset_k$ и всеми $itemset_{k\text{-}1}$. Например (см. рис. 2), для набора $itemset_3 = \{i_1 i_2 i_4\}$ поднаборами будут $itemset_2 = \{\{i_1 i_3\}, \{i_1 i_4\}, \{i_3 i_4\}\}$, а разностями $R = \{\{i_4\}, \{i_3\}, \{i_1\}\}$ соответственно, тогда все импликации $A \to B$ будут выглядеть, как $A \to B = \{(\{i_1 i_3\} \to \{i_4\}), (\{i_1 i_4\} \to \{i_3\}), (\{i_3 i_4\} \to \{i_1\})\}$. При этом импликация $A \to B$ будет относится к АП тогда и только тогда, когда $supp(A \to B) > minsupp$ и $conf(A \to B) > minconf$.

На этапе поиска популярных наборов можно выделить два процедуры: генерация наборов и расчет поддержки набора. Первые алгоритмы поиска популярных наборов (AIS и SETM) генерировали наборы и рассчитывали поддержку во время чтения транзакций из ТБД, не используя при этом свойство антимонотонности [6].

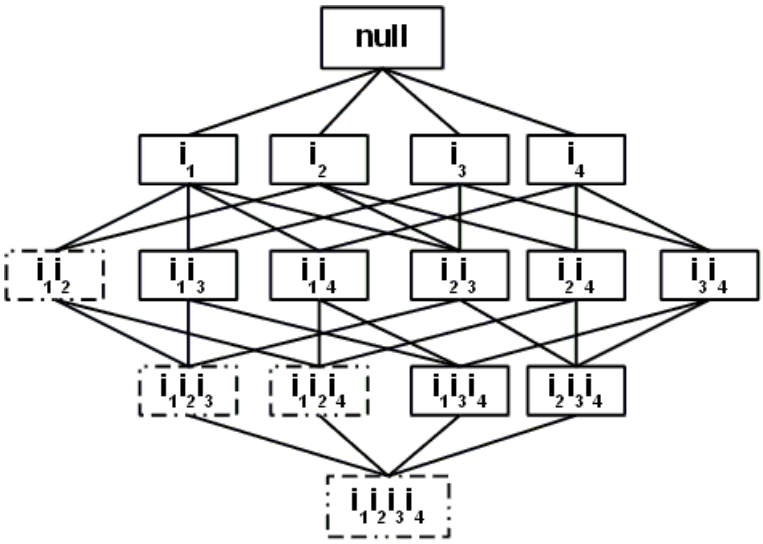

Рис. 2. Решетка связей возможных наборов элементов из ТБД для $I = \{i_1, i_2, i_3, i_4\}$

Сокращение времени поиска популярных наборов можно добиться за счет использования алгоритма *Apriori*. Работа данного алгоритма состоит из некоторого числа (проходов) повторяющихся процедур генерации *k*-элементных наборов-кандидатов (*candidate generation*) и подсчета поддержки наборов-кандидатов (*candidate counting*). При этом генерация кандидатов заключается в создании множества *k*-элементных кандидатов (где *k* – номер этапа) в результате чтения транзакций из ТБД. В отличие от алгоритмов AIS и SETM поддержка при этом не рассчитывается. При подсчете поддержки кандидатов вычисляется поддержка каждого *k*-элементного набора-кандидата и выполняется удаление кандидатов, поддержка которых меньше *minsupp*. Оставшиеся *k*-элементные наборы считаются популярными. В табл. 2, 3 и 4 показаны результаты работы первого, второго и третьего прохода алгоритма *Apriori* с 1-, 2- и 3-элементными наборами-кандидатами.

Количество повторяющихся проходов *k* алгоритма *Apriori*, как правило, равно количеству элементов в самом длинном наборе. В данном примере самым длинным является 4-элементный набор (см. табл. 1) для транзакции с пятым номером: $TID = 5$, $k = 4$. Однако, поддержка ни одного из 3-элементных наборов-кандидатов не больше *minsupp*, поэтому процесс поиска популярных наборов завершается уже после 3-го прохода, и можно переходить ко второму этапу анализа связей в ТБД – разработки АП на основе найденных популярных наборов.

Для разработки АП будут использованы только 2-элементные популярные наборы, т.к. уже отмечалось, что $supp(\bullet)$ ни одного из 3-элементных наборов-кандидатов не больше *minsupp* (см. табл. 4).

На основе 2-элементные популярных наборов сформируем все возможные импликации типа $A \to B$ и рассчитаем их поддержку и достоверность, результаты запишем в табл. 5. Согласно табл. 5, для $minconf = 60\%$ к АП можно отнести следующие правила: «если фото, то онлайн»; «если фото, то альбом»; «если фото, то рамки»; «если фото, то магазин».

Таблица 3.

Транзакционная база данных 2-элементных наборов в нормализованном виде

Элементы	$i_3 i_4$	$i_2 i_4$	$i_2 i_5$	$i_2 i_8$	$i_3 i_4$	$i_3 i_5$	$i_3 i_8$	$i_4 i_5$	$i_4 i_8$	$i_5 i_8$
Значения	онлайн фото	онлайн альбом	онлайн рамки	онлайн магазин	фото альбом	фото рамки	фото магазин	альбом рамки	альбом магазин	рамки магазин
1	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	1	0	0	0	0	0
3	1	1	0	0	1	0	0	0	0	0
4	0	0	0	0	0	1	0	0	0	0
5	1	0	1	0	0	1	0	0	0	0
6	1	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	1	0	0	0
8	1	0	1	0	0	1	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	1	0	0	0
11	0	0	0	0	1	0	1	0	1	0
12	0	0	0	0	0	1	1	0	0	1
13	0	0	0	0	1	1	0	1	0	0
14	0	0	0	0	0	0	0	0	0	0
supp ($i_k i_j$)	28.6	7.14	14.3	0	28.6	35.7	28.6	7.14	7.14	7.14
minsupp 28.6%	$itemset_2$				$itemset_2$	$itemset_2$	$itemset_2$			

Таблица 4.

Транзакционная база данных 3-элементных наборов в нормализованном виде

Элементы	$i_2 i_3 i_4$	$i_2 i_3 i_5$	$i_2 i_3 i_8$	$i_3 i_4 i_5$	$i_3 i_4 i_8$	$i_3 i_5 i_8$	$i_4 i_5 i_8$
Значения	онлайн фото альбом	онлайн фото рамки	онлайн фото магазин	фото альбом рамки	фото альбом магазин	фото рамки магазин	альбом рамки магазин
1	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0
3	1	0	0	0	0	0	0
4	0	0	0	0	0	0	0
5	0	1	0	0	0	0	0
6	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0
8	0	1	0	0	0	0	0
9	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0
11	0	0	0	0	1	0	0
12	0	0	0	0	0	1	0
13	0	0	0	1	0	0	0
14	0	0	0	0	0	0	0
supp ($i_k i_j i_l$)	7.14	14.3	0	7.14	7.14	7.14	0
minsupp 28.6%							

Методика разработки семантического ядра сайта на основе алгоритма *Apriori*

Реализация двухэтапного анализа связей для поиска популярных наборов с помощью алгоритма *Apriori* и разработки АП на основе найденных популярных

наборов нашла свое отражение в аналитическом приложении *Deductor Academic*, версия 5.2, которого была использована для создания методики разработки СЯС.

Методику разработки СЯС можно представить в виде следующих шагов:

1) Формирование ТБД поисковых запросов с помощью средств статистики поисковых систем.

2) Конвертация ТБД поисковых запросов в текстовый формат.

3) Создание нового сценария в аналитическом приложении *Deductor Academic*.

4) Использование «Мастера импорта» для импорта ТБД в текстовом формате в приложение *Deductor*.

5) Использование «Мастера экспорта» для экспорта результатов обработки в текстовом формате.

6) Использование «Мастера обработки» для реализации «Ассоциативных правил», как одного из методов «Data Mining».

7) Формирование результирующей ТБД на основе текстового файла.

8) Перезапись атрибута *content* мета тэг *keywords* на основе результирующей ТБД.

Таблица 5.

Наборы-кандидаты в АП типа импликации $A \to B$

$itemset_2$	$itemset_1$	R	$A \to B$	$supp(A \to B)$, %	$conf(A \to B)$, %
$\{i_2\ i_3\}$ онлайн - фото	$\{i_2\}$ онлайн	$\{i_3\}$ фото	онлайн → фото	28.6	33
	$\{i_3\}$ фото	$\{i_2\}$ онлайн	фото →онлайн	28.6	80
$\{i_3\ i_4\}$ фото - альбом	$\{i_3\}$ фото	$\{i_4\}$ альбом	фото → альбом	28.6	100
	$\{i_4\}$ альбом	$\{i_3\}$ фото	альбом → фото	28.6	33
$\{i_3\ i_5\}$ фото - рамки	$\{i_3\}$ фото	$\{i_5\}$ рамки	фото → рамки	35.7	100
	$\{i_5\}$ рамки	$\{i_3\}$ фото	рамки →фото	35.7	41.6
$\{i_3\ i_8\}$ фото - магазин	$\{i_3\}$ фото	$\{i_8\}$ магазин	фото → магазин	28.6	100
	$\{i_8\}$ магазин	$\{i_3\}$ фото	магазин → фото	28.6	33

При этом термины «Мастер импорта», «Мастер обработки», «Ассоциативные правила», «Мастер экспорта» являются специфическими для приложения *Deductor* [3].

Предлагаемая методика была реализована для разработки СЯС типа Интернет-витрина Konica-Digital. При этом результат заключительного шага – формирование атрибута *content* мета тега *keywords*, будет выглядеть так: `<meta name="keywords" content="интернет магазин, магазин интернет, купить онлайн, фотографии онлайн, фотографии печать, фотографии рамки, магазин рамки интернет, купить онлайн рамки, ...">`.

Выводы

Реализация предлагаемой методики разработки СЯС с динамическим контентом позволила поднять позиции Konica-Digital в SERP на 25% для 70% информационных, 85% транзакционных и 60% нечетких запросов, вводимых пользователем в основные поисковые системы Yandex и Google. При этом в 1.5 раза сократились затраты рабочего времени специалиста по SEO, необходимые для достижения заявленных результатов.

Ограниченный объем статьи не позволил показать другие приложения методики разработки СЯС с динамическим контентом. Однако необходимо заметить, что при реализации предлагаемой методики для Интернет-магазина Vsedetali в качестве ТБД использовалась таблица заказов, а автоматизированное формирование атрибута *content* мета тэгов *keywords* на основе АП также позволило повысить полноту и точность, снизить время разработки семантического ядра сайта. Таким образом, предложенная методика разработки СЯС является достаточно универсальной, и с небольшими

доработками может быть применена для эффективного продвижения сайтов с динамическим контентом специалистами по SEO.

Список литературы

1. Ашманов И.С. Оптимизация и продвижение сайтов в поисковых системах / И.С. Ашманов, А.А. Иванов. – 3-е изд. – СПб. : Питер, 2011. – 463 с.
2. Как работают поисковые системы – сниппет, алгоритм обратных индексов, индексация страниц, особенности работы поисковиков [Электронный ресурс] / Режим доступа: http://ktonanovenkogo.ru/seo/search/kak-rabotayut-poiskovye-sistemy-snippet-index.html
3. Паклин Н.Б. Бизнес-аналитика: от данных к знаниям / Н.Б. Паклин, В.И. Орешков. – СПб.: Питер, 2009. – 624 с.
4. Chung D. Suchmaschinen-Optimierung: Der schnell Einstieg / D. Chung, A. Klünder. – Heidelberg : REDLINE/mitp, 2007. – 224 S.
5. Aden T. Google Analytics: Implementieren. Interpretieren. Profitieren. – Auflage: 2., aktualisierte und erweiterte Auflage. – München : Carl Hanser Verlag, 2010. – 463 S.
6. R. Agrawal, T. Imielinski, A. Swami. Mining Associations between Sets of Items in Massive Databases // Proc. of the ACM-SIGMOD 1993 Int'l Conference on Management of Data, Washington D.C., May 1993. – PP. 207-216.

РОЗРОБКА СЕМАНТИЧНОГО ЯДРА САЙТУ З ДИНАМІЧНИМ КОНТЕНТОМ НА ОСНОВІ АСОЦІАТИВНИЙ ПРАВИЛ

О.О. Арсірій, О.О. Ігнатенко, О.О. Леус

Одеський національний політехнічний університет,
просп. Шевченка, 1, Одеса, 65044, Україна; e-mail: o-ignatenko@mail.ru

В результаті аналізу проблем просування в пошукових системах веб-ресурсів з динамічним контентом запропонована методика розробки семантичного ядра сайту на основі створення асоціативних правил за допомогою алгоритму пошуку популярних наборів Apriori в транзакційній базі даних пошукових запитів. Застосування методики дозволило підвищити повноту і точність, а також знизити час розробки семантичного ядра сайту типу Інтернет-вітрини та магазину.

Ключові слова: пошукова система, пошукові запити, семантичне ядро сайту, популярні набори, асоціативні правила, алгоритм Apriori

DEVELOPMENT OF A SEMANTIC CORE OF A WEB-SITE WITH DYNAMIC CONTENT BASED ON ASSOCIATION RULES

Elena A. Arsiry, Olga A. Ignatenko, Alexei A. Leus

Odessa National Polytechnic University,
1 Shevchenko Ave., Odessa, 65044, Ukraine; e-mail: o-ignatenko@mail.ru

In consequence of an analysis of promotion problems of web sites with dynamic content in the search engines a methodology of development of a web-site's semantic core was proposed namely through the association rules creation by using the search of popular sets in the transactional search queries' database – algorithm Apriori. Application of the methodology has allowed to improve the completeness and accuracy, and reduce the time on the development of a semantic core for such web-sites as an Internet-showcase and shop.

Keywords: search engine, search queries, semantic core of a web-site, popular sets, association rules algorithm Apriori