УДК 615.47:616-085

Г.В. ПЕВЦОВ, д-р техн. наук (г. Харьков),
Э.Г. ФАСТОВСКИЙ, НТУ "ХПИ",
М.А. ОЛЕЙНИК, НТУ "ХПИ"

АНАЛИЗ МЕТОДОВ КОНСОЛИДАЦИИ ИНФОРМАЦИИ И ОСОБЕННОСТЕЙ ИХ ПРИМЕНЕНИЯ

У статті розглянуті найбільш поширені моделі пошуку інформації, що засновані на різних математичних методах, які можуть бути покладені в основу системи консолідації інформації. Проведено їх аналіз, виявлені їх недоліки.

The most widespread information retrieval models, based on different mathematical methods which can be put in a basis of the consolidation information system, are considered in this article. Their analysis is conducted, their lacks are revealed.

Постановка проблемы. Объем информации, производимой человеком, постоянно увеличивается. С каждым днем все актуальней становится проблема ее обработки. Помочь в этом может консолидация информации. Термин "консолидация" может иметь много значений. В широком смысле под консолидацией можно понимать процесс поиска, отбора, анализа, структурирования, преобразования, хранения, регистрации (каталогизации) и предоставления потребителю информации по заданным темам. Основными этапами консолидации информации могут быть:

– добывание информационных сведений;

– обработка сведений и получение информационных данных;

– преобразование и обобщение информационных данных, получение, структурирование и сохранение информации по заданной теме.

Для добывания информационных сведений могут применяться следующие методы:

– поиск источников информации;

– наблюдение за источниками информации.

Поиск источников информации – целенаправленные действия по выявлению источников информации, определению их основных характеристик (достоверности, непротиворечивости, своевременности и др.) и анализу смысла распространяемых (передаваемых) сведений.

Наблюдение – целенаправленные действия по систематическому добыванию информационных сведений от выявленных источников путем отбора и регистрации распространяемых (передаваемых) сведений. Наблюдение может быть непрерывным, периодическим, контрольным.

Основные требования (критерии эффективности) к системе консолидации информации: непрерывность; активность; целенаправленность; своевременность; достоверность представляемой информации.

Консолидация информации в узком смысле – это процесс отбора

информации по заданной теме. Если рассматривать консолидацию в этом значении, то к системам, осуществляющим консолидацию, можно отнести информационно-поисковые системы (ИПС). В ИПС применяются автоматические алгоритмы принятия решения о соответствии документов заданной теме. Наработки в сфере поиска информации могут быть использованы в качестве отправной точки при разработке алгоритмов и методов консолидации информации.

Анализ литературы. На сегодняшний день существует множество моделей поиска, основанных на разнообразных математических методах, на базе различных вариаций которых строятся современные ИПС. Известные сегодня модели поиска можно разделить на три группы [1]:

– теоретико-множественные (логические) – булева модель, модель нечетких множеств, расширенная булева модель [1 – 6];

– алгебраические – векторная модель [7 – 9], модель, основанная на латентно-семантическом анализе [10 – 11];

– вероятностные модели [12].

Приведенный перечень не является исчерпывающим. В каждой из указанных групп существует множество моделей. В виду невозможности (и не целесообразности) охватить все существующие модели в статье анализируются только наиболее востребованные в ИПС.

Цель статьи – анализ наиболее распространенных моделей поиска информации, которые могут быть положены в основу системы консолидации информации.

Булева модель (строгая булева модель). Булева модель – наиболее простая, часто применяемая интуитивная модель поиска. Она основана на использовании аппарата теории множеств и математической логики. В модели используются операции теории множеств, такие как объединение, пересечение и вычитание множеств. Суть модели состоит в следующем [2]. Для массива документов, среди которых требуется осуществлять поиск, строиться матрица, называемая терм-на-документ матрицей. Матрица представляет собой отношение между документами и термами индексирования. Строки матрицы соответствуют термам документов, столбцы представляют собой поисковые образы документов. Если на пересечении строки i и столбца j стоит 1, то это означает, что терм i встречается в документе j. Поисковый запрос пользователь составляет из ключевых слов, соединенных между собой с помощью логических операций $\cap$, $\cup$, $\neg$. С помощью логических операций возможно построить запрос любого уровня сложности, ограничить расстояние между ключевыми словами, исключить из результатов поиска документы, в которых присутствуют указанные пользователем слова и т.д.

К достоинствам булевой модели можно отнести ее простоту и относительную легкость реализации, вследствие чего она применяется во

многих поисковых системах. Недостатков у этой модели значительно больше. Во-первых, сложно построить запросы для большинства пользователей, поскольку они должны быть хорошо знакомы с булевой алгеброй. Во-вторых, в результат поиска попадут только те документы, в которых встречаются все слова из запроса пользователя: документ, в котором отсутствует терм из запроса, но присутствует его синоним, найден не будет. В-третьих, полученные документы невозможно ранжировать, поскольку все термы в каждом документе считаются одинаково важными. В-четвертых, в данной модели плохо масштабируем поиск [3]. Оператор $\bigcap$ может очень сильно сократить число документов, которые выдаются на запрос, поскольку при отсутствии в документе хотя бы одного терма запроса он будет признан нерелевантным, как и документ, не содержащий ни одного из них. Оператор $\bigcup$, напротив, может привести к неоправданно широкому запросу, в котором полезная информация затеряется за информационным шумом, поскольку документ, содержащий только один из термов запроса, будет признан релевантным, как и документ, содержащий все термы запроса. И, наконец, размеры результатов поиска невозможно контролировать.

Модель нечетких множеств. Данная модель базируется на теории нечетких множеств [4, 5], которая допускает частичную принадлежность элемента множеству в отличии от традиционной теории множеств, не допускающей этого. В данной модели весь массив документов описывается как набор нечетких множеств термов. Каждый терм определяет некую монотонную функцию принадлежности к документам данного массива. В данной модели переопределяются логические операторы для учета возможности частичной принадлежности множеству. Пользовательские запросы выражаются так же, как и в булевой модели, и обрабатываются подобным образом с использованием переопределенных логических операторов. Логический оператор $\bigcap$ интерпретируется как минимум из двух функций (MIN), соответствующих термам запроса, оператор $\bigcup$ – как максимум (MAX), оператор $\neg$ – как (1 – <значение функции>). Используется интервал [0, 1] вместо множества [0, 1] как в булевой модели. Вес терма может быть определен следующим образом:

– 0, если терм отсутствует в документе;

– $0.5 + 0.5 \times TF/MAX - TF$, если терм присутствует в документе, где TF – частота данного слова в данном документе, MAX – переопределенный оператор $\bigcap$;

– некоторое приведенное значение, если вместо данного терма присутствует родственный ему терм.

Достоинство модели – возможность ранжировать результаты.

Недостатки. В запросе вида $A \bigcap B \bigcap C \bigcap D$ будет считаться только терм с наименьшим значением, а в запросе вида $A \bigcup B \bigcup C \bigcup D$ – только терм с

наивысшим значением; вычислительные затраты и затраты на дисковое пространство для хранения выше, чем у булевой модели. Также модель страдает от недостаточной способности различать результаты поиска практически в той же степени, что и булева модель. Модель не позволяет назначать веса термам пользовательского запроса.

К положительным сторонам рассмотренных булевой модели и модели нечетких множеств можно отнести то, что они требуют меньших объемов вычислений (при индексировании и определении релевантности отобранных документов запросу) по сравнению с другими моделями. Они менее алгоритмически сложные и предъявляют менее жесткое требование к дисковому пространству для хранения поисковых образов документов.

Расширенная булева модель (*p*-norm модель) является модификацией булевой модели [6], в которой предпринята попытка снять ограничения, присущие булевой модели. По сути, она является гибридом булевой модели и рассматриваемой ниже векторной модели. В ней предполагается, что терм описывает содержимое документа с некоторой точностью, которая выражается в виде веса терма. Для определения веса терма используется статистика встречаемости терма с соответствующей процедурой нормализации. Причем определять веса можно как для термов документа, так и для термов запроса.

В данной модели документ представляется вектором $\vec{d}$ в пространстве [5], охватывающем множество ортонормированных векторов термов. Подобие поискового запроса и документа определяется путем обобщенного скалярного произведения соответствующих векторов в пространстве документа. Обобщение использует *p*-нормы, определенные для *n*-мерного вектора $\vec{d}$, длина которого определяется формулой [5]

$$\left|\vec{d}\right| = \left|(w_1, w_2, ..., w_n)\right| = \left(\sum_{j=1}^{n} w_j^p\right)^{\frac{1}{p}}, \qquad (1)$$

где $1 \le p \le \infty$, $w_1, w_2, ..., w_n$ – веса термов, входящих в документ, представленный вектором $\vec{d}$.

Для *p*-норм модели определены обобщенные логические операторы $\bigcap$ и $\bigcup$. Интерпретация запроса может быть изменена путем использования различных значений p при вычислении подобия запроса и документа. Когда $p = 1$, то различие между логическими операторами $\bigcap$ и $\bigcup$ исчезает.

Если все термы запроса взвешены одинаково и $p = \infty$, то интерпретация запроса такая же, как и в модели нечетких множеств. Наоборот, если термы запроса не взвешены и $p = \infty$, то модель ведет себя как строгая булева модель. Варьируя значения p от 1 до ∞ можно определить модель, чье поведение соответствует точке на шкале от векторной модели до модели нечетких

множеств и строгой булевой модели. Чем выше значение p, тем строже становятся логические операторы. Экспериментально установлено, что наилучшее значение p лежит в диапазоне от 2 до 5.

Преимуществами данной модели являются возможность использования структурированных запросов и взвешенных термов (что позволяет ранжировать полученные в результате поиска документы), а также – менять интерпретацию структуры запроса через варьирование значения *p*.

Недостатки модели состоят в том, что на практике ее сложно реализовать, поскольку необходимо применить дополнительный алгоритм взвешивания термов. Модель трудна для понимания пользователям.

Векторная модель предложена Дж. Сэлтоном [7]. В ней документы и запросы пользователей представляются в виде *n*-мерных векторов в *n*-мерном векторном пространстве. Размерность векторного пространства *n* – это общее количество различных термов во всех документах. Все разнообразие словоформ каждого терма приводится к некоторой основе. В рассмотрении не участвуют так называемые "стоп-слова" – служебные, малоинформативные и высокочастотные словоформы. Каждому терму t_i документа D_j (и запроса Q) ставится в соответствие неотрицательный вес w_{ij} (w_i для запроса). Каждый документ и запрос представляются соответствующими векторами:

$$\vec{D}_j = (w_{1j}, w_{2j}, ..., w_{nj}),$$

$$\vec{Q} = (w_1, w_2, ...w_n).$$

Способ определения значения весов w_{ij} в модели не определен, но он имеет большое влияние на поисковую эффективность ИПС. Для определения веса терма применяется множество различных методов. Это, например, может быть частота встречаемости терма в документе TF (term frequency), определяемая по формуле:

$$tf = \frac{m_j}{\sum_{k=1}^{n} m_k},$$

где m_j – количество появлений рассматриваемого терма в документе, а знаменатель – сумма количеств появления всех слов в документе.

Если известна частота встречаемости терма во всех документах коллекции, то применяется TF-IDF мера для определения веса, где IDF (inverse document frequency) – обратная частота документа. Она может быть вычислена, например, по следующей формуле [8, стр. 463]:

$$IDF = \log \frac{N}{D_i}, \qquad (2)$$

где N – число документов в коллекции, D_i – число документов коллекции, в которых встречается терм t_i. Функция (2) приписывает большие веса тем термам, которые встречаются лишь в нескольких документах.

Имея два вектора $\vec{D}_j$ и $\vec{Q}$ можно определить их подобие, значение которого и будет определять степень релевантности документа D_j запросу Q. Это значение применяется для ранжирования найденных документов – если оно больше некоторой пороговой величины, то документ считается релевантным, и наоборот. Способ определения подобия двух векторов в модели также не определен. Обычно угол между векторами используется как мера расхождения векторов, и косинус угла используется как числовое подобие (поскольку косинус обладает хорошим свойством быть равным 1 для одинаковых векторов и равным 0 для ортогональных векторов) [9]. Также в качестве подобия может использоваться скалярное (внутреннее) произведение векторов, манхэттенское расстояние, расстояние Чебышева и т.д.

Преимущества данной модели заключаются в ее простоте и возможности ранжирования результатов.

Недостатки модели:

– существует сложность определения логического отрицания $\neg$, т.е. исключения из результатов поиска документов, содержащих определенный терм;

– модель требует больших вычислительных затрат, поскольку требует обработки больших объемов данных;

– плохо поддерживается синонимия, т.е. документы считаются далекими друг от друга, если у них нет совпадающих слов.

Модель, основанная на латентно-семантическом анализе. В теории информационного поиска данную модель принято называть латентно-семантическим индексированием. Латентно-семантический анализ (Latent Semantic Analysis – LSA) – это метод извлечения и представления контекстно-зависимых значений слов с помощью статистической обработки больших наборов текстовых документов [10]. LSA был запатентован в 1988 году. Впервые модель была предложена в работе [11]. В этой модели совокупность всех контекстов, в которых встречается и не встречается данное слово, задает множество взаимных ограничений, которые позволяют определить подобие смысловых значений слов и множеств слов между собой.

LSA состоит из двух стадий – обучения и анализа проиндексированных данных. На стадии обучения исходные документы представляются в виде матрицы X, строки которой соответствуют термам, а столбцы – документам. Элемент матрицы (i, j) содержит частоту появления терма t_i в документе d_j. Далее к матрице X применяется метод SVD (Singular Value Decomposition) – разложение матрицы по сингулярным значениям. Такое разложение может осуществляться с помощью различных алгоритмов, например, с помощью

алгоритма Якоби, алгоритма Голуба-Кахана-Рейнча. Разложение матрицы по сингулярным значениям дает в результате три матрицы *D, T,* и *S*, такие что $X = DST^t$. У матриц *D* и *T* столбцы ортонормированные, а *S* – диагональная матрица сингулярных значений.

Такое разложение обладает следующим свойством: если в *S* оставить только k наибольших сингулярных значений, а в матрицах *D* и *T* только соответствующие этим значениям столбцы, то произведение получившихся матриц будет наилучшим приближением исходной матрицы *X* матрицей ранга k : $X \approx X' = D'S'T'$. Это приводит к значительному уменьшению размерности исходного пространства (обычно k находится в диапазоне от 50 до 400). Таким образом, каждый терм и документ представляется вектором меньшей размерности.

Вторая стадия представляет собой оценивание подобия между парой документов, парой слов или между словом и документом. Определить подобие между любой комбинацией термов и/или документов можно теми же способами, что и между векторами в векторной модели (например, с помощью скалярного произведения векторов).

Преимущества модели по сравнению с векторной состоят в том, что используется пространство векторов значительном меньшей размерности, чем в векторной модели; модель позволяет решать проблему синонимии. Модель не требует никаких предварительных действий, в том числе вмешательства человека.

Однако модель не лишена недостатков. Ее сложность приводит к тому, что она зачастую проигрывает по скорости другим моделям. Выбор значения размерности k является серьезной проблемой – если k слишком велико, то модель по своим характеристикам приближается к векторной модели, при малом значении k не удается установить различия между похожими словами и документами.

Вероятностные модели. Данные модели поиска основаны на применении методов теории вероятности. В них используются статистические показатели, определяющие вероятность релевантности документа поисковому запросу пользователя. В основе этих моделей лежит принцип вероятностного ранжирования (Probabilistic Ranking Principle, PRP) [12]. Суть этого принципа заключается в том, что документы в коллекции должны ранжироваться по убыванию вероятности их релевантности запросу пользователя. Вычисление этой вероятности является ключевой частью модели, и этим большинство вероятностных моделей отличается друг от друга. Первоначально идея таких моделей была предложена в работе [13].

Особенности модели заключаются в следующем. В модели учитываются взаимозависимости и связи термов, и определяются такие основные параметры, как веса термов запроса и форма подобия "запрос-документ". В основе модели лежат два главных параметра: *ррел и рнерел*, которые являются,

соответственно, вероятностью релевантности и нерелевантности документа пользовательскому запросу. Они вычисляются на основе вероятностных весов термов и фактического присутствия термов в документе. Подразумевается, что вероятность является бинарным свойством, поэтому *ррел*=1-*рнерел*. Также в модели применяются два стоимостных параметра, которые определяют потери, связанные с включением в результат нерелевантного документа и пропуском релевантного.

Взвешивание термов базируется на основе вероятностного подхода, который предполагает наличие обучающего набора документов, релевантных пользовательскому запросу. Этот набор может быть сформирован из числа наиболее релевантных документов, полученных в результате выполнения запроса с использованием модели поиска, такой как векторная модель. Документы для обучающего набора, как правило, отбираются самим пользователем.

Начальный запрос определяется как набор термов. Определенное число наиболее релевантных документов в отношении начального запроса используется для формирования обучающего набора. Для вычисления веса терма оцениваются следующие условные вероятности с использованием обучающего набора: документ релевантен запросу при условии, что терм встречается в документе; и документ не релевантен запросу при условии, что терм появляется в документе [14].

Пусть N – общее число документов, из которых R релевантно пользовательскому запросу; R_t – число релевантных документов, содержащих терм t, который появляется в f_t документах. Оцениваются условные вероятности:

– P {t присутствует в документе | документ релевантный} = R_t / R;

– P {t присутствует в документе | документ нерелевантный} = = $(f_t - R_t) / N - R$;

– P {t отсутствует в документе | документ релевантный} = $R - R_t / R$;

– P {t отсутствует в документе | документ нерелевантный} = = $((N - R) - (f_t - R_t)) / (N - R)$.

Из этих оценок вес w_t терма t может быть выведен с использованием формулы Байеса:

$$w_t = \log \frac{R_t / (R - R_t)}{(f_t - R_t) / (N - f_t - (R - R_t))}.$$

Числитель (знаменатель) выражает вероятность появления терма t в релевантном (нерелевантном) документе. Веса терма, большие 0, показывают, что появление терма в документе является свидетельством релевантности этого документа запросу. Значения весов, меньшие 0, показывают противоположное.

Недостатки модели заключаются в низкой вычислительной масштабируемости и необходимости постоянного обучения системы, качество поиска уступает векторным моделям.

Выводы. Таким образом, в статье рассмотрен ряд моделей, применяемых в настоящее время для отбора информации по заданной теме. Все модели (за исключением булевой) обеспечивают ранжирование полученных документов. Ранжирование в теории позволяет расположить первыми наиболее релевантные запросу документы. Однако на практике это обеспечивается не всегда. Даже если в документе встречаются все термы из запроса, то это не означает, что документ удовлетворит потребность пользователя, хотя он и является релевантным. Если же в документе содержатся не все термы запроса, то это в свою очередь не означает его нерелевантность запросу (особенно если пользователь желает получить любую информацию по указанной теме). В данном случае в любой модели интерес представляет оценивание того, насколько такой документ соответствует запросу. Именно этот вопрос является узким местом любой модели. Порог, который разделяет области принятия решений о соответствии или несоответствии такого документа запросу, выбирается или интуитивно, или в результате проведения большого числа экспериментов. Выбор этого порога остается открытой исследовательской проблемой.

Список литературы: **1.** *Сегалович И.* Как работают поисковые системы // Сайт поисковой системы Яndex.ru. **2.** *Ланкастер Ф.У.* Информационно-поисковые системы. – М.: Мир, 1972. – 308 с. **3.** *Храмцов П.* Информационно-поисковые системы Internet // Открытые Системы. – 1996. – № 3. **4.** *Radecki T.* Fuzzy Set Theoretical Approach to Document Retrieval // Information Processing and Management. – 1979. – № 15 (5). – P. 247–259. **5.** *Venkat N. Gudivada, Vijay V. Raghavan, William I. Grosky, Rajesh Kasanagottu*. Information Retrieval on the world wide web // IEEE Internet Computing. – 1997. – № 9–10. – P. 58–68. **6.** *Salton G., Fox E., Wu H.* Extended Boolean information retrieval // Communications of the ACM. – 1983. – № 26 (11). – P. 1022–1036. **7.** *Salton G, Wong A, Yang. C.* A Vector Space Model for Automatic Indexing // Communications of the ACM. – 1975. – № 18 (11). – P. 613–620. **8.** *Солтон Дж.* Динамические библиотечно-информационные системы. – М.: Мир, 1979. – 558 с. **9.** *Amit Singhal* Modern Information Retrieval: A Brief Overview // Data Enginering Bulletin, IEEE Computer Society. – 2001. – V. 24, – № 4. – P. 35–43. **10.** *Landauer T., Foltz P., Laham D.* An introduction to latent semantic analysis // Discourse Processes. – 1998. – № 25. – P. 259–284. **11.** *Deerwester S., Dumais Susan, Furnas G. W., Landauer T. K., Harshman R.* Indexing by Latent Semantic Analysis // Journal of the American Society for Information Science. – 1990. – № 41 (6). – P. 391–407. **12.** *Robertson S. E.*. The probabilistic ranking principle in IR // Journal of Documentation. – 1977. – № 33. – P. 294–304. **13.** *Maron M. E., Kuhns J.L.* On relevance, probabilistic indexing and information retrieval // Journal of the ACM. – 1960. – № 7. – P. 216–244. **14**. *Robertson S.E., Sparck K. Jones*. Relevance weighting of search terms // Journal of the American Society for Information Science. – 1976. – № 27 (3). – P. 129–146.

Поступила в редакцию 28.09.2007