УДК 519.175

ОПТИМАЛЬНАЯ ОРГАНИЗАЦИЯ МОРФОЛОГИЧЕСКИХ СЛОВАРЕЙ В ПОИСКОВЫХ СИСТЕМАХ

А. П. Якубенко, К. Е. Селезнев, М. А. Артемов

Воронежский государственный университет

В работе описывается принцип создания модуля для построения и использования морфологических словарей в информационно-поисковых системах. Проводится анализ подходов к проблеме, исследуется производительность полученной системы. Доказана сходимость предлагаемого итерационного процесса построения словаря.

В данный момент существует множество систем для индексирования и поиска электронной документации. В последние годы отмечается резкое увеличение объёмов текстовой информации, хранимой в электронном виде [1]. Среди причин этого явления следует указать, во-первых, переход большого числа предприятий и организаций на электронный документооборот и электронное делопроизводство, во-вторых, построение различных электронных библиотек, в-третьих, создание электронных текстовых архивов (прежде всего, юридических документов), и, наконец, развитие сети Интернет. Это привело к созданию всевозможных автоматических систем обработки неформализованных текстовых документов [2]. К таким системам относятся, прежде всего, информационно-поисковые системы (ИПС), предоставляющие возможности поиска необходимых текстовых документов по условиям, налагаемым на содержимое этих документов и указываемых в подаваемых на вход ИПС запросах. Следует отметить, что наибольшее развитие получили системы поиска информации в сети Интернет: информационно-поисковые системы, электронные библиотеки, автоматические текстовые классификаторы, фильтры и др.

Для поиска в этих системах используются модули морфологического, синтаксического, семантического анализа текстовых документов, словари тезауруса. Типичная архитектура работы информационно-поисковой системы представлена на рис. 1.

Морфологический анализ является базовым для всех остальных модулей. Скорость и качество работы синтаксического, семантического анализа, поиска отношений в словарях тезауруса напрямую зависит от скорости и эффективности морфологического словаря — наиболее часто используемая его операция в перечисленных механизмах — получение канонической (базовой, основной) формы слова, проверка на принадлежность двух словоформ одному базовому слову, приведение получение производной словоформы из канонической.

Таким образом, к основным задачам модуля морфологического анализа выбранного естественного языка относятся:

— Получение основной словоформы и морфологических характеристик указанной словоформы.

— Построение словоформы с указанными морфологическими характеристиками по основной словоформе.

— Возможность навигации по списку всех возможных словоформ языка.

Требования, предъявляемые к данной конкретной реализации модуля морфологического анализа — высокая скорость поиска словоформ и получения всей информации по ним. Минимальная скорость определяется необходимостью индексирования больших документов (порядка 10000 слов) за время порядка 1 сек.

Наполнение словаря предполагается производить используя словарь Зализняка [3]. На размер словаря ограничений не накладывается, предполагаемое количество словоформ словаря — порядка 8000000. Т.е. требуется построить систему эффективного поиска из 8 млн словоформ со скоростью доступа около 0.0001 секунды. Существенны также ограничения на количество используемой оперативной памяти. В связи с тем, что модуль будет использован не как отдельное приложение, а в составе сложной системы, которой также необходимы большие объемы оперативной памяти для быстрой работы (кэширование страниц баз данных, сортировки данных), модуль морфологического анализа не должен использовать более 10 мБ оперативной памяти.

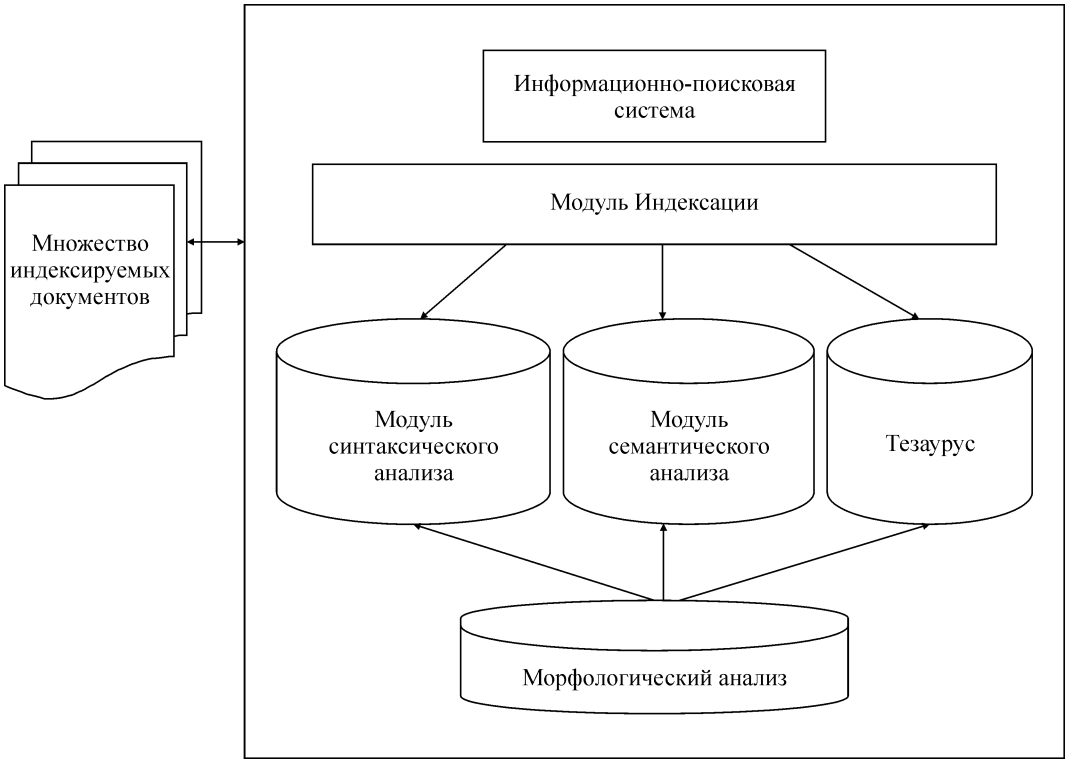

Рис.1. Структура информационно-поисковой системы

Одним из вариантов получения словоформ по исходному слову является синтез словоформ на основе правил языка. Но, данный подход имеет ряд недостатков.

В общем случае, правила формообразования естественных языков, и в частности русского, достаточно сложны для программной реализации т.к. имеют большое число исключений [3]. В связи с этим, включить все возможные правила построения словоформ в данный модуль не представляется возможным ввиду многообразия вариантов формообразования зачастую даже у одной морфологической характеристики, а также наличия огромного количества частных случаев, не подчиняющихся стандартным алгоритмам.

Неплохие результаты дают системы анализа построенные на семантических сетях [4]. Но, результат работы таких систем зависит от качества обучения, и, в общем случае не дает гарантий точности образования форм. Чаще всего результатом его работы является набор вероятностей принадлежности определенной словоформы к некоторым базовым. Кроме того, для выделения канонической формы анализируемого слова необходимо иметь исчерпывающий словарь словоизменительных основ (один из вариантов реализации). Данный словарь может быть получен автоматически путем морфологического разбора большого массива слов (обучающая выборка) и отсечения у них окончаний. Тем не менее, даже такой словарь оказывается чрезмерно емким (~1 млн основ). Наличие такого емкого словаря ограничивает возможности семантического анализатора по производительности, поскольку процедура поиска словоизменительной основы канонической формы носит итеративный характер, связанный с перебором гипотетически возможных остальных морфем, где на каждой итерации выполняется обращение к емкому словарю основ. Как показали исследования, более рациональным подходом является хранение списка полных словоформ, что приводит к увеличению размеров словаря, но резко повышает производительность, избавляя от итеративного подхода перебора основ при поиске основной словоформы.

Поэтому, наиболее общим является подход создания словарей. Все возможные словоформы, разбиваются на отдельные группы, образуемые словоформами одного и того же слова. В каждой группе выделяется основная словоформа и остальные словоформы. Таким образом, *словом* оказывается целая группа словоформ этого слова. В каждом словаре при построении заданы как сами морфологические характеристики и их

возможные значения, так и множество всех словоформ с соответствующей им информацией о приписанных им значениях граммем.

Существующие открытые электронные словари не являются полными. Более того, в литературе часто появляются новые термины (например, в сфере высоких технологий), поэтому словарь необходимо постоянно пополнять. Общеизвестно, что для получения высокой производительности любой системы, необходима узкая ориентация на определенные виды действий.

Модуль морфологического анализа является системой, в основном ориентированной на анализ, то есть 99 % времени работы модуль выполняет одну из указанных выше операций поиска и только 1 % времени уходит на управление самим модулем (задание, наполнение, корректировка словаря). Поэтому, морфологический словарь хранится в двух видах:

— В неоптимизированном виде, позволяющем проводить модификацию словаря.

— В оптимизированном виде, не позволяющем производить модификацию, но обеспечивающем высокую скорость выполнения операций поиска. Оптимизированный вид получается путём конвертации словаря из неоптимизированного вида.

Модуль анализа предназначен для применения в системах полнотекстового поиска в СУБД ЛИНТЕР и в ПК СПЭД, разработанных в ЗАО НПП «РЕЛЭКС» [5].

СУБД ЛИНТЕР, помимо возможностей полноценной реляционной СУБД, при выполнении запросов к базе данных позволяет указывать ограничения на отдельные слова, формирующие содержимое текстовых данных (поля типа CHAR, VARCHAR, BLOB и т.д.). В условиях, предъявляемым к текстовому содержимому, может быть указано:

— требование наличия указанного слова;

— требование наличия слов, начинающихся с указанных символов;

— требование наличия слов, заканчивающихся указанными символами;

— требование наличия слов, содержащих указанные символы;

— требование наличия слов, похожих на указанное в запросе (под похожестью понимается различие в одном или нескольких символах);

— логические комбинации из перечисленных типов требований.

Специальный способ индексирования обеспечивает максимальную скорость выполнения запросов. В настоящий момент подсистемой полнотекстового поиска СУБД ЛИНТЕР поддерживаются основные распространённые форматы хранения текстовых документов (txt, html и т.д.), но при необходимости перечень поддерживаемых форматов может быть расширен.

Использование модуля морфологического анализа позволяет сравнивать словоформы, встречающиеся в тексте, с учётом их основной формы, то есть учёт морфологии даёт возможность, например, считать слова *модель* и *моделями* одним и тем же словом, за счёт чего существенно повышается эффективность поиска информации в СУБД.

Программный комплекс семантического поиска электронных документов (ПК СПЭД) ориентирован на индексацию и эффективный поиск русскоязычных текстовых документов, в ходе которого используется морфологическая, синтаксическая, семантическая и статистическая информация о текстовых документах.

В ходе своей работы ПК СПЭД использует модуль морфологического анализа в целях индексации и поиска текстовых документов, синтаксического анализа, выявления устойчивых словосочетаний, для определения ключевых слов и словосочетаний текстовых документов.

Основной проблемой при реализации модуля хранения оптимизированных словарей является структура хранения данных. Данные словаря должны быть представлены в виде набора файлов на диске. Необходимым условием является возможность портирования модуля и словарей на различные платформы (MS Windows, Linux, МСВС и т.д.).

В качестве структуры хранения рассматривались:

— Использование существующей базы данных. Недостатками этого подхода является большой объем получаемой базы данных (более 1Гб при 8000000 словоформ), и низкая производительность при получении информации — 100000 SQL-запросов выполняется порядка 5 минут. Это является неплохим результатом для данных в других приложениях, но медленно для текущей задачи.

— Простой список словоформ с индексной информацией. Данный способ хранения не обеспечивает ни компактности, ни скорости доступа (сложный поиск нужного слова).

— Архивированный список словоформ. Данный способ не обеспечивает скорости доступа.

— Список словоформ и их отличий. Предполагается хранить только отличия окончаний текущего слова от предыдущего. Как показал эксперимент, данный способ не обеспечивает ни скорости доступа (необходимо проследить достаточно длинную цепочку для формирования слова, даже при наличии т.н. «ключевых слов»), ни компактности ввиду многообразия слов естественного языка.

— Хранение информации в виде TRIE-дерева. TRIE-структуры (словари, боры) — класс структур данных, в которых значение ключа, по которому производится поиск, рассматривается как упорядоченная последовательность (строка) знаков из некоторого алфавита, содержащего V символов [6]. Само название "TRIE" происходит от слова "reTRIEval" (поиск, выБОРка). Как показали исследования, такой способ реализации является наиболее оптимальным для данных требований. Информация хранится в виде сильно ветвящегося дерева по буквам слов. Для ускорения поиска и возможности распараллеливания поиска словоформы разбиваются на группы — по начальной букве слова. В узлах, соответствующих окончанию слова, хранится информация об установленных граммемах и их значениях.

Размеры всех полей, необходимые для хранения данных словаря, определяются динамически, по максимальному размеру данных, которые могут в них храниться.

Реализация этого потребует дополнительных усилий — побитной работы с файлом, но позволит избежать хранения незначащих нулевых битов и даст компактность хранения информации.

Файл имеет страничную организацию. В целях повышения производительности, размер страницы кратен размеру кластера при хранении данных в целевой файловой системе.

В целях минимизации обращения к файловой системе, модуль анализа имеет собственную систему кэширования страниц. Система имеет оригинальную стратегию вытеснения страниц, а также двухуровневый КЭШ — для каждого словаря в отдельности, а также общий большой КЭШ. Как показали эксперименты, размер КЭШа увеличивает общую производительность только до определенного уровня. Общий КЭШ является более эффективным, чем отдельный для каждого словаря. С определенного момента увеличения выделенного объема памяти начинается даже незначительное замедление работы — на поиск страницы требуется больше времени, чем на получение её из файла. Оптимальным оказался размер КЭШа порядка сотен страниц файла словаря.

В связи с тем, что для уменьшения суммарного размера словаря и повышения производительности словоформы были сгруппированы по начальной букве, возникла необходимость объединения данных. Данную функцию выполняет модуль генерального словаря. Особенностью данного модуля является возможность объединения словарей с различной морфологической структурой. Это необходимо для возможности дальнейшего расширения — возможности добавления новых характеристик хранимых данных. Добавляемые словари могут иметь расширенную морфологическую структуру по сравнению с имеющимися словарями — это не потребует перегенерации имеющихся словарей для добавления в них новых морфологических или любых других признаков (например, обозначение устаревших форм, специфических терминов). Генеральный словарь не хранит в своих данных структуры подключенных словарей, что несколько замедляет скорость работы с моделями словарей и ведет к дополнительной нагрузке на файловую систему, но данные обращения к файлам словарей быстро кэшируются модулем анализа и не сказываются на производительности. Вместе с тем, недостатком такого подхода является то, что обязанность по контролю над структурой словарей ложится на клиентское приложение. Возможна ситуация, когда необходимого свойства нет в словаре. В этом случае модуль анализа отработает корректно (доступа к недопустимым страницам файлов не произойдет), и клиентскому приложению будет возвращена ошибка по несоответствию структуры.

Было произведено тестирование модулей морфологического анализа, создания неоптимизированных и конвертации их в оптимизированную форму. Установлено, что модули имеют высокую производительность. Основная характеристика — скорость доступа к лексемам словарей. В качестве тестового примера был использован случайный набор словоформ размером 100000 слов. Часть из них присутствовала в словаре, часть — отсутствовала (в написа-

ние были внесены ошибки). Под управлением операционной системы MS Windows 100000 запросов к модулю анализа были выполнены за 4 секунды. Под управлением операционной системы МСВС на аналогичном оборудовании данный тест был выполнен за 2,5 секунды.

Качество работы системы оценивалось также по количеству обнаруженных словоформ. Для текстов из предметной области, на которую ориентирован ПК СПЭД, процент обнаружения словоформ составил 85 %. В случае обнаружения словоформы качество определения грамматических значений было близко к 99 % (в соответствии со словарем Зализняка). Неточности и многозначность трактовок появлялись только в случае омонимов и омоформ. Было произведено сравнение с системой морфологического анализа Автоматическая Обработка Текста «Диалинг» (www.aot.ru). Обнаружены ошибки в системе «Диалинг» — несоответствия в построении повелительного наклонения. Разработанная система работала в несколько раз (до 10) быстрее АОТ.

Скорость генерации оптимизированных словарей находится на приемлемом уровне. Генерация полного словаря занимает несколько часов. Столь долгий процесс генерации объясняется тем, что модуль строит полную структуру словарей сначала в памяти, затем вычисляет оптимальные размеры полей, и, если они изменились с прошлого построения, перестраивает словарь. Это связано с тем, что от вычисленных необходимых размеров полей зависит суммарный объем словаря. Таким образом, процесс построения словарей является итерационным.

Сходимость данного процесса не очевидна. Рассмотрим простейший случай. Предположим, что файл словаря полностью состоит только из одного поля, длина которого N бит (параметр корректируется в процессе построения словаря). Пусть данное поле должно является ссылкой на конкретный байт файла (это граничный случай — «наихудший» при страничной организации структуры файла размер страницы полагаем равным 1 байту).

Предположим, что таких полей в файле находится M штук. Т.е. суммарный объем файла будет равен M*N/8 байт. Т.е. любое число, которое меньше либо равно M*N/8 должно умещаться в N бит. Получаем формулу

$$2^N >= (M*N/8).$$

Известно, что степенная функция 2^N растет быстрее линейной a*N, где a=M/8 (для доказательства этого факта достаточно продифференцировать обе функции по переменной N — для степенной функции все производные будут положительны, а для линейной — уже вторая производная будет равна 0).

Таким образом, итерационный процесс сходится. В точном решении данного неравенства нет необходимости, т.к. файл имеет более сложную структуру, чем простой набор из M чисел.

Предполагаемая архитектура использования разработанных модулей представлена на рис. 2.

Непосредственно в клиентское приложение встраиваются только модули анализа и модуль редактирования словарей. Остальные модули программы поставляются как утилиты.

При изменении структуры модуль пригоден не только для хранения морфологических словарей, но и для других целей (тезаурус и т.п.).

В работе предложен способ создания быстрого морфологического словаря. Приведены результаты тестирования разработанных модулей. При необходимости разработки систем морфологического анализа можно воспользоваться предлагаемым подходом. Показана сходимость предложенного процесса построения словаря. Достигнута основная цель — разработан алгоритм, реализация модулей которого удовлетворяет поставленным условиям. Полученные модули протестированы на производительность и пригодны для использования. К недостаткам реализации можно отнести достаточно большой объем получаемых словарей. В связи с этим необходимы дальнейшие модификации для уменьшения объема оптимизированных словарей.

Предлагаемый подход позволяет получать довольно точные результаты, вместе с тем недостатком подхода является отсутствие результатов поиска и невозможность получения канонической формы при отсутствии словоформы в исходном словаре. В этом случае поисковая система вынуждена производить поиск не по всем словоформам, а по конкретной данной, что снижает возможности поиска. Частично эту проблему можно решить при помощи предлагаемого неоптимизированного словаря, который позволяет дополнять существующий словарь. Предложен механизм генерального словаря, позволяющий рассматривать набор различных по структуре словарей как единый словарь.

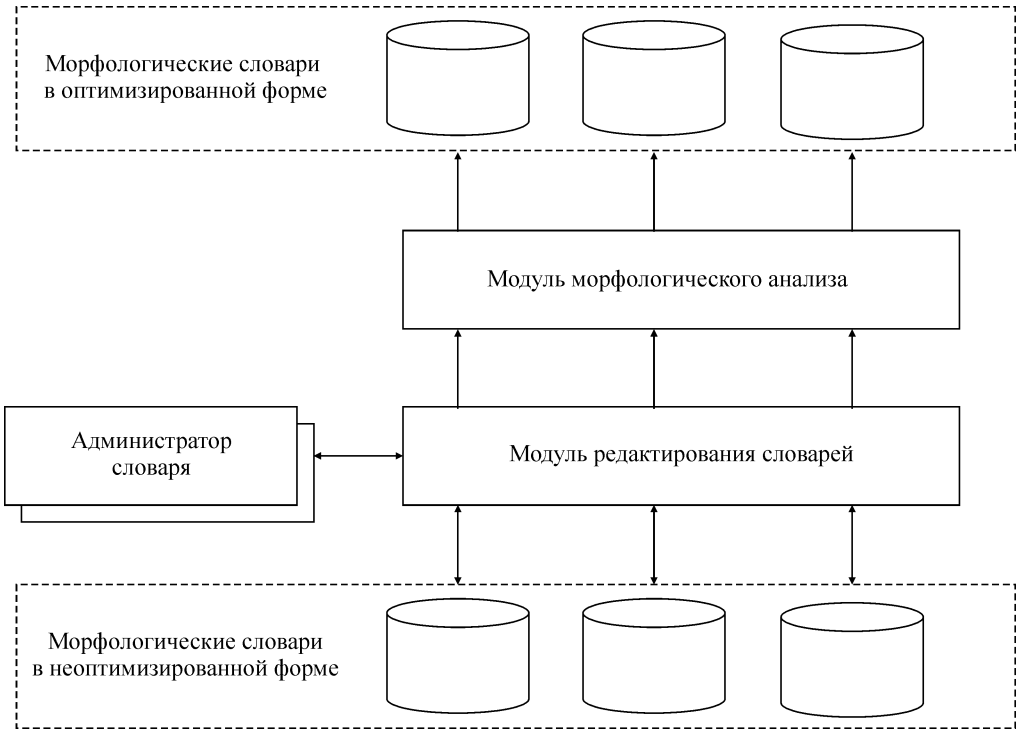

Рис. 2. Архитектура использования модулей морфологического анализа

В сравнении с другими имеющимися системами, разработанная система гарантирует точное определение характеристик в случае обнаружения словоформы, имеет высокую производительность. Недостатком является отсутствие результата в случае отсутствия словоформы в словаре.

СПИСОК ЛИТЕРАТУРЫ

1. *Селезнёв К.Е.* Системы поиска / Открытые системы. — 2005. — № 10.

2. *Белоногов Г.Г.* Компьютерная лингвистика и перспективные информационные технологии / Г. Г. Белоногов Ю. П. Калинин, А. А. Хорошилов // М.: Русский мир, 2004.

3. *Зализняк А.А.* Грамматический словарь русского языка (словоизменение) / 2-е изд. — М.: Русский язык. — 1980.

4. *Сегалович И.* Русский морфологический анализ и синтез с генерацией моделей словоизменения для не описанных в словаре слов / И. Сегалович, М. Маслов // http://company.yandex.ru/articles/article1.html

5. Описание СУБД ЛИНТЕР http://www.relex.ru/linter_rus.php

6. *Тиори Т.* Проектирование структур баз данных: В 2-х кн. / Т. Тиори, Дж. Фрай // Кн. 2. М.: Мир. — 1985. — 320 с.