2В

УДК 004.934

Д.В. Выдрин, В.Н. Поляков
Московский государственный институт стали и сплавов, Россия

Реализация электронного словаря с использованием n-грамм

Многие существующие словари используют модели, основанные на морфологии описания слов. Унификация морфов позволяет получить компактное и довольно быстрое решение. В качестве минимальной единицы для описания слова могут выступать слоги или набор аффиксов, которые сопоставлены с каким-либо корнем. В данной работе предложен вариант организации словаря с использованием псевдоморфов – n-грамм.

Введение

Системы обработки документов на естественном языке, как правило, используют разнообразные словари. Существует несколько подходов и способов организации электронных словарей [1]. Большинство из них связано с моделями хранения данных, основанных на различных видах организации морфов или аффиксов.

Для электронного словаря важнейшими свойствами являются скорость анализа входной последовательности символов и компактность. Эти характеристики особенно важны при использовании словаря при обработке больших объемов текста.

Словари, использующие для поиска хэширование (закрытое или открытое) [2], обычно основаны на предварительном разборе входного слова: отброс аффиксов, выявление графической основы. После чего осуществляется сопоставление и проверка наличия аффиксов для выделенной основы, если она найдена в словаре. При этом процедура поиска значительно усложняется, что сказывается на сложности программного кода. При этом, однако, скорость обработки получается очень высокой.

В подходах с использованием древовидных структур [3] при поиске осуществляется переход по ветвям дерева в зависимости от положения букв во входной цепочке (конечный автомат). При этом сохраняются скорости, не уступающие хэш-таблицам. А сам словарь занимает в памяти столько же места, сколько и на диске.

Назначение словаря определяет его информативное наполнение, т.е. дополнительную информацию, хранимую для каждой словарной статьи. Это может быть грамматическая информация о слове, толкование слова, ссылки на синонимы или перевод. Зачастую разделение словарей по функциям приводит и к разным способам реализации в разных программных продуктах.

Реализация словаря

В своей работе мы задавались целью создания более обширной базы лингвистических данных для того, чтобы программное приложение словаря позволяло решать разнотипные задачи лингвистики. Такие, например, как:

- индексация текста и морфоанализ,
- извлечение грамматических характеристик слов,
- восстановление всей словоизменительной парадигмы для отдельного слова,
- подбор кандидатов для орфокоррекции,
- поиск схожих слов,
- генерация новых псевдоформ или ассоциация гипотетических словоформ с какой-либо основой,
- получение списка толкований, синонимов, перевода, лексического окружения, а также других всевозможных объединений слов по различным признакам.

Задача описания русской морфологии удачно решена А.А. Зализняком в 80-е годы [4]. Более того, существуют действующие модели описания (хранения) стандартных слов и слов-исключений, основанных на словаре А.А. Зализняка [5]. По сути дела, морфологический анализ русского языка предполагает хранение для каждой графической основы ссылки на соответствующую парадигму склонения или спряжения. Однако из соображений универсальности словаря для решения перечисленных задач в предлагаемом словаре сделано отступление от общепринятых моделей. В частности, предполагается, что словарь может расширяться. При этом любое новое слово должно быть обработано и сохранено так, как оно дано в тексте-первоисточнике или внесено экспертом-лексикографом.

При решении задачи индексации текста каждому входному слову нужно поставить в соответствие уникально характеризующий его идентификатор. Исходная форма слова, или лемма, не может служить уникальным идентификатором слова из-за омонимии. В связи с этим в словаре имеется возможность считать две одинаковые по написанию леммы разными. Остальные возможности поддержки объединения слов основываются на хранении ссылок у каждой уникальной леммы.

Словарь разделен на три отдельные структуры. Первая структура отвечает за поиск входной цепочки букв и состоит из хэш-таблицы адресующей к словарным статьям. Применен метод открытого хэширования слова по сигнатуре [6].

Вторая структура – массив основ с соответствующими ссылками на описание словоизменения характерных для данной основы. За основу представления словаря были взяты n-граммы (цепочки длины n). Для каждой буквы русского алфавита было сделано отображение на целое число ‘а’ – 1 , ‘б’ – 2, ‘в’ – 3 и т.д. (за исключением буквы ё, которая читается как е). Кодирование одной буквы требует 5-ти бит (для алфавита из 32-х букв). Таким образом, цепочку из 3-х букв, занимающую 3 байта, можно уместить в 15 битах или типичной структуре word, равной 2-м байтам. Этот прием позволил не только хранить вместо трех байт два, но и перейти на другой уровень программной логики: от одного символа – к триаде. Использование отображения на число позволило вычислить еще одну специфичную характеристику тестовой последовательности – сумму всех букв, которую чаще называют сигнатурой.

Итак, каждая основа в словарной статье была разбита на триады (цепочки длины 3 или меньше) которые занимают в памяти 2 байта и могут быть однозначно раскодированы.

Дополнительно для каждой основы хранилось: ссылка на запись в таблице окончаний (2 байта), идентификатор постоянной грамматической характеристики (1 байт), ссылка на перевод (3 байта), синоним(3 байта).

Третья структура – таблица словоизменения, массив триад и непостоянных грамматических характеристик.

Для наполнения словаря используется специальная программа-транслятор, которая переводила входной текст в формат словаря. На входе транслятора – тестовый файл с записями, содержащими развернутую парадигму слова, грамматические характеристики и перевод. Грамматические характеристики закодированы следующим образом. Первый параметр (1 байт) – основная характеристика части речи (например, одушевленность и род у существительных, как правило, не меняются), второй параметр (1 байт) – изменяемая характеристика (например, падеж, число). Первое слово в этом списке считается леммой. Транслятор анализирует входной массив, отбирает оптимальный вариант графической основы (основ), заполняет массивы основ и окончаний, с ограничением, что длина отброшенного графического суффикса не может превысить трех букв, чтобы сохранить триаду. Допустим, есть набор слов «я, ты, он, она, они» такая последовательность осмысливается как набор основ с окончаниями я+<null>, ты+<null> он+<null,а,и>. Для уменьшения количества основ процедура минимизации наделена принципом выявления неменяющихся аффиксов (например, частицы «-ся»), а также выявлением случая пропуска буквы (клин-ок, клинк-а, в этом случае слово «клинок» соотносится с основой «клинк-» с псевдоокончанием «о», у которого стоит реверсивная помета, означающая перестановку «о» с «к» при выводе слова «клинок»). Занесение информации о синонимах или переводе может быть выполнено в ручном или автоматическом режиме – из файла.

Дополнительный набор внутренних ссылок используется для хранения более сложных структур. Например, слово «ковер-самолет» не хранится в словаре явно. У леммы «ковер» стоит дополнительная внутренняя ссылка на лемму «самолет» с пометой на то, что эти слова пишутся через тире, и оба слова изменяются. Количество видов таких помет ограничено числом 255, и для каждой пометы программно описана процедура ее обработки. Максимальное количество самих ссылок составляет 16 млн. (24 бита). В дальнейшем таким способом предполагается хранить двукоренные слова (например, электростанция, автопоезд и т.п.), «счетные» прилагательные (например, шестнадцатипудовый, сорокотонный) и некоторые устойчивые словосочетания.

Трансляция слов, полученных из 82 тыс. лемм из словаря А.А. Зализняка, занимает около 4-х минут. Сохранение массивов словаря около – 2-х секунд. Загрузка массивов около одной секунды. Ниже приведены общие характеристики словаря без дополнительных модулей, таких, как перевод или данные о синонимах, которые могут подключаться отдельно.

Заключение

Характеристики словаря на 82 тысячи слов (лемм).

Объем словаря (лемм) – 82 тысячи.
Количество основ – 199 380.
Занимаемое место на диске – 6,8 Мб.
Занимаемый объем памяти:
– хэш-таблица – 3 Мб (16-битный ключ).
– словарь основ – порядка 4,5 Мб.
– таблица окончаний – 0,3 Мб.
– дополнительные переменные – 0,2 Мб.
Время трансляции – 4 минуты.
Время сохранения – 2 секунды.
Время загрузки – 1 секунда.
Скорость лемматизации – 20 тысяч слов в секунду (170Кб/с).
Возможно добавление и удаление словарных статей без перезагрузки словаря.

Общие ограничения модели словаря:

Максимальное число основ – 16777215 (приблизительно 10 млн. лемм).
Максимальное число схем окончаний – 65534.
Максимальная длина входного слова – 24 символа.
Максимальный размер входной парадигмы – 200 слов.
Максимальное количество окончаний у одной леммы – 60.
Только русский алфавит (без «ё»).
Все слова переводятся в нижний регистр.
Максимальное количество дополнительных видов ссылок – 200.

В настоящее время основным применением электронного словаря является блок морфологического анализа и орфокоррекции в проекте «Интеллектуальная поисковая машина» [7]. Тестовая версия лемматизатора, созданного на основе данного словаря, доступна в сети Интернета по адресу: http://macrocosm.narod.ru/download/morfull.rar

Литература

1. Сегалович И. Реализация словаря на основе разряженой хэш-таблицы // Труды Междунар. семинара «Dialog`95». – Таруса. – 1995.
2. C++ Primer Answer Book by Bruce Leung & Clovis Tondo Addison-Wesley. – May, 1999.
3. http://www.linguist.ru
4. Зализняк А.А. Грамматический словарь русского языка. – М., 1977.
5. Аношкина Ж.Г. Морфологический процессор русского языка // Говор. – Сыктывкар, 1995.
6. Бойцов Л.М. Использование хеширования по сигнатуре для поиска по сходству // Прикладная математика и информатика. – 2001. – № 8.
7. Поляков В.Н. Интеллектуальная поисковая машина. Концептуальный проект // Труды Казанской школы по компьютерной и когнитивной лингвистике «TEL – 2000». – Вып. 5. – Казань: Сэлэт. – 2000.

Common of existing electronic dictionaries use models, based on word's morphology. A morph's unification allow to get compact and effective solution. Syllables or affixes, which confronted with some root are as minimum unit for the word's models. A dictionary organization variant with using n-gram as pseudomorph is suggested in this project.

Статья поступила в редакцию 03.07.02.