OPC-trie: спецификация оптимального классификатора для СУБД НИКА

В.А. Тищенко[I,II]

[I] Федеральное государственное учреждение «Федеральный исследовательский центр «Информатика и управление» Российской академии наук», г. Москва, Россия
[II] Образовательное частное учреждение высшего образования «Православный Свято-Тихоновский гуманитарный университет», г. Москва, Россия

Аннотация. Постулируется, что PATRICIA-trie является способом построения многоуровневого индекса, наряду со стандартным способом построения индекса в виде алфавитного списка ключей в БД НИКА. Представлена схема описания данных для многоуровневого индекса, который строится для индексного атрибута. Определяется оптимальное сжатое по путям префиксное дерево OPC-trie. OPC-trie рассматривается как спецификация алфавитного классификатора для вершин типа массив БД НИКА.

Ключевые слова: *PATRICIA-trie, OPC-trie, спецификация оптимального алфавитного классификатора.*

DOI: 10.14357/20790279210108

1. Актуальность оптимального алфавитного классификатора

Развитие информационных технологий дает толчок для дальнейшей разработки пользовательского интерфейса. Появление мобильных устройств позволяет делать акцент на гипертекстовом интерфейсе для баз данных. Пользователь выбирает ключ и переходит на искомый объект. Такой сценарий работы приводит к необходимости организации алфавитных классификаторов для массивов ключей в БД. При объемах массивов более тысячи ключей такой классификатор становится необходимым элементом гипертекстовой системы [1]. Префиксное дерево trie [2], образованное из букв или сочетаний букв, является отправной точкой для алфавитного классификатора.

2. Построение префиксного дерева стандартными средствами БД НИКА

Модель БД НИКА является типовополной [3]. Это означает, что с помощью этой модели можно представить объекты со структурой любой сложности. В частности префиксное дерево можно представить средствами БД НИКА в виде рекурсивного шаблона. Таким образом, чтобы построить индекс для текстового атрибута не в виде списка ключей, упорядоченного по алфавиту, а в виде многоуровневого индекса на основе PATRICIA-trie [4], необходимо в схеме описания данных в соответствующем массиве под вершиной INDEX добавить рекурсивный шаблон «Буквы». На рис.1 приведен пример данного шаблона для индекса по полю «ФИО». Рекурсивный шаблон ветвления применяется в точках ветвления PATRICIA-trie (сжатое префиксное дере-

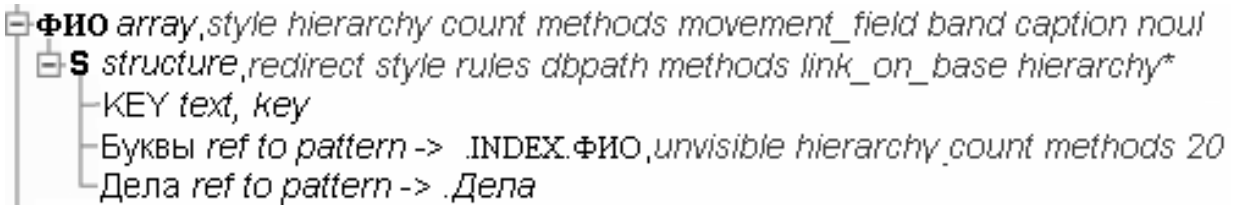

Рис.1. Фрагмент схемы описания данных для многоуровневого индекса по полю «ФИО» с методами отображения вершин БД

во без однопутевых ветвей). При этом в ключ KEY на каждом уровне шаблона «Буквы» записывается соответствующий префикс из букв, составляющих часть ключа исходного массива.

На рис. 2 представлен пример префиксного дерева для иллюстрации схемы описания данных на рис.1. В скобках приводятся для данного префикса число непосредственно подчиненных префиксов и число подчиненных листьев. На первом уровне в качестве ключа записывается начальный префикс исходного ключа. На последнем уровне, который находится под конечным префиксом исходного ключа, создается массив по шаблону, соответствующему массиву, в котором находится атрибут, по которому строится индекс. В рассматриваемом примере на рис. 1 – это массив «Дела» по шаблону, соответствующему корневому массиву «Дела». Построенный многоуровневый индекс полностью соответствует PTRICIA-trie для данного массива ключей (с учетом уровня массива «Буквы»).

3. Применение комбинированного метода, сочетающего префиксное дерево и список ключей

Э. Сассенгат в статье [5] предложил сочетать в цифровом поиске префиксное дерево trie со списком ключей путем прерывания ветвления на определенном уровне trie. При этом происходит переход на список ключей, который соответствует префиксу, пройденному по пути в trie. Искомый ключ ищется в списке ключей на выбранный префикс. Сассенгат использовал префиксное дерево для получения списка из шести ключей. Однако такой выбор числа ключей в классе не дает в общем случае оптимального классификатора. Задача сводится к определению вершин, в которых производить сжатие префиксного дерева и прерывать ветвление для перехода на список ключей для получения оптимального классификатора в смысле числа переходов по префиксам и группам префиксов или ключей и числа просмотров префиксов и ключей.

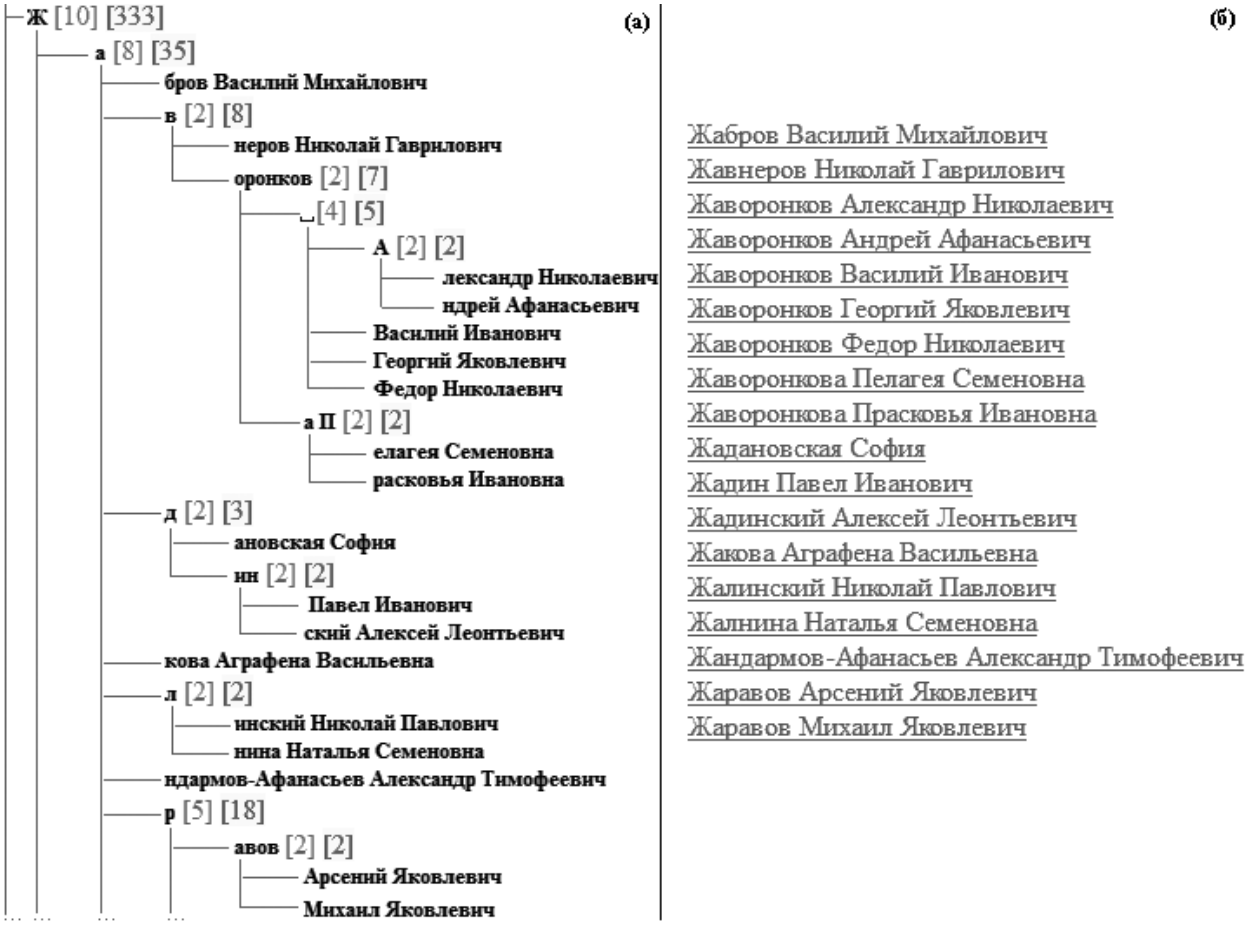

Рис.2. Фрагмент префиксного дерева для многоуровневого индекса по полю «ФИО» (а), с соответствующим ему списком ключей (б)

4. Использование функционала общего числа операций для получения параметров оптимального классификатора

Развитием метода Сассенгата является получение оптимального trie в смысле общего числа операций просмотра и переходов по префиксам и ключам при перемещении к искомому ключу. При этом префиксное дерево PATRICIA-trie можно дополнительно сжимать, объединяя различные вершины в один префикс. Для нахождения оптимальной структуры префиксного дерева вводятся параметры, с помощью которых на основе префиксного дерева строится классификатор: максимальное число ключей в классе n и число ключей в группе n_g, причем $n_g \le n$. В работах [6,7] через неявную зависимость от n и зависимость от n_g выражается функционал общего числа операций в префиксном дереве $S_{\text{оп}}(n,n_g)$. Используя $S_{\text{оп}}(n,n_g)$, OPC-trie (optimal path compressed trie) или оптимальное сжатое по путям префиксное дерево определяется как префиксное дерево с параметрами n^* и n_g^*.

Опр.1. OPC-trie=trie(n^*,n_g^*), где $(n^*,n_g^*) = \arg\min\{S_{\text{оп}}(n,n_g)\}$, $S_{\text{оп}}(n^*,n_g^*)=S_{\text{оп}}^*$ – оптимальное значения функционала общего числа операций, которое определяется в смысле [7].

Опр.2. Оптимальный классификатор – это OPC-trie с полными префиксами.

Полный префикс берется от корня префиксного дерева до текущего префикса включительно.

5. Построение оптимального классификатора на основе OPC-trie

В равномерном случае число уровней классификатора получается как $\log_{n^*} N$-1, где $n^*=a^k$, a=|A| и A – алфавит, N – число ключей в массиве. В неравномерном случае можно рассмотреть префиксное дерево и объединять ключи и префиксы в классы, начиная с листьев дерева, поднимаясь вверх к корню. Нижний уровень OPC-trie образуют те вершины префиксного дерева, в которых число листьев максимально и не превышает n^*. Следующий, более верхний уровень OPC-trie, образуется путем объединения префиксов нижнего уровня в классы, в которых число префиксов максимально и не превышает n^*. Процедура объединения префиксов в классы завершается верхним уровнем OPC-trie, который содержит один класс префиксов, число которых не более n^*. Каждый класс префиксов или ключей, образующий уровень в классификаторе, делится на группы по $n_g^* \le n^*$ префиксов или ключей.

6. Применение оптимального классификатора

На рис.3 приводится пример второго уровня оптимального классификатора. Первый уровень представляет собой однобуквенные префиксы. Классификатор строился для индекса по полю «ФИО» объемом примерно 36 тыс. ключей. В докладе [8] приводится пример данных для поля

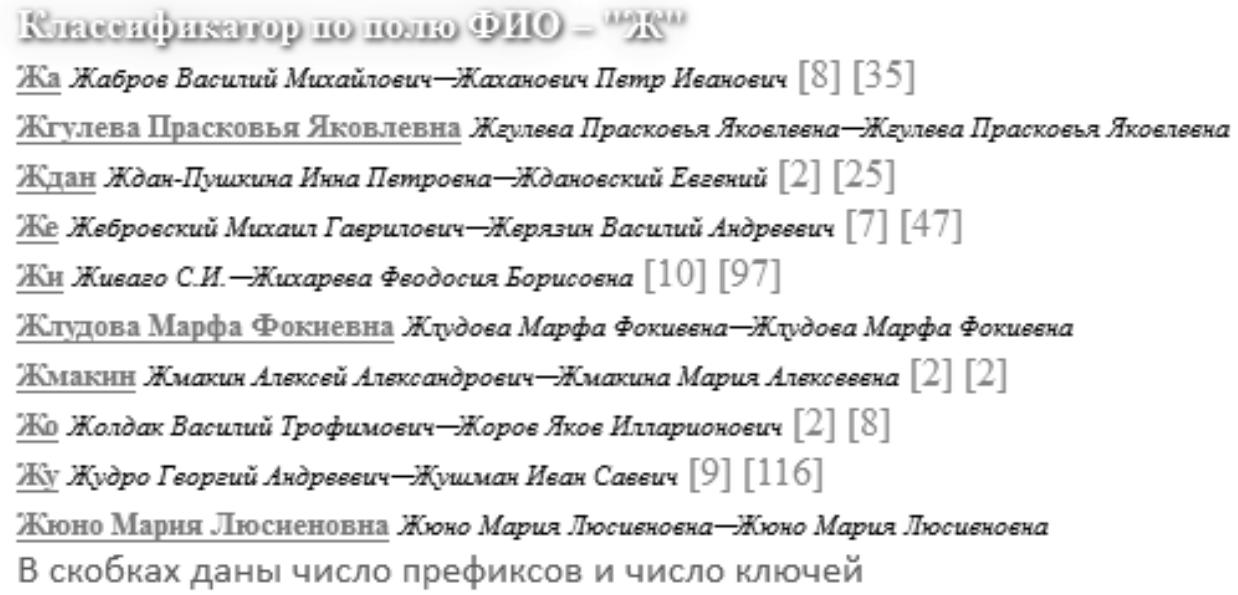

Рис.3. Пример уровня оптимального классификатора на префикс «Ж»

«ФИО» и значения функционала $S_{оп}(n,n_g)$, который имеет характерный минимум. Существование минимума $S_{оп}(n,n_g)$ также показывают и другие данные разных объемов массивов ключей. Это не является доказательством в математическом смысле, но лишь наводит на гипотезу о существовании такого минимума $S_{оп}{}^{*}=S_{оп}(n^{*},n_g{}^{*})$. Величины n^{*} и $n_g{}^{*}$ представляют собой значения параметров классификатора, которые задаются как входные параметры функции, реализующей спецификацию, отображающую PATRICIA-trie в виде ОРС-trie или оптимального алфавитного классификатора.

Литература

1. *Емельянов Н.Е., Тищенко В.А.* Представление гипертекста в СУБД НИКА // Технология программирования и хранения данных / Труды ИСА РАН. 2009. Т.45. С. 17-36.
2. *Briandais R*. File Searching Using Variable Length Keys / R. Briandais // Proc. AFIPS Western Joint Computer Conference, San Francisco, California, USA, 15, March 1959. P. 295-298.
3. *Годунов А.Н.* СУБД НИКА / А.Н. Годунов, Н.Е. Емельянов, А.Н. Косьмынин, В.А. Солдатов // Системы управления базами данных и знаний. М.: Финансы и статистика, 1991. С.209-249.
4. *Morrison D.* PATRICIA-practical algorithm to retrieve information coded in alphanumeric / D. Morrison // J. ACM 15,4(0ct. 1968). P. 514-534.
5. *Sussenguth E.H.* Use tree structures for processing files / E.H. Sussenguth // CACM 6. 1963. P.272-279.
6. *Тищенко В.А*. Выбор оптимального алфавитного классификатора при минимизации общего числа операций // Труды ИСА РАН. 2018. Т. 68. № 1. С. 54-57.
7. *Арлазаров В.Л.* Устройство отыскания информации по ключевым словам / В.Л. Арлазаров, В.А. Тищенко // Патент на изобретение № 2679967 С1 Российская Федерация. 2019. Бюл. № 5.
8. *Тищенко В.А.* Реализация классификатора по лексикографическому признаку для ООСУБД НИКА / В.А. Тищенко // Материалы XXXIII Международной научно-практической конференция “Eurasiascience”. 15 ноября 2020 г. С.69-71.

Тищенко Владимир Александрович. Федеральное государственное учреждение «Федеральный исследовательский центр «Информатика и управление» Российской академии наук», г. Москва, Россия. Научный сотрудник. Кандидат философских наук. Количество печатных работ: 24. Область научных интересов: средства создания и поддержки электронных библиотек и электронных изданий. E-mail: vtischenko@isa.ru

OPC-trie: specification of the optimal classifier for the NIKA DBMS

V.A. Tishchenko[I,II]

[I] Federal Research Center "Informatics and control" of The Russian Academy of Sciences, Moscow, Russia

[II] St. Tikhons"s Orthodox University, Moscow, Russia

Abstract. It is postulated that PATRICIA-trie is a method to build a multilevel index, along with the standard method to build an index as an alphabetical list of keys in the NIKA database. A schema for definition data for a multilevel index, which is built for an index attribute, is presented. The optimal compressed OPC-trie prefix tree is determined. OPC-trie is considered as a specification of an alphabetical classifier for vertices of the NIKA database array type.

Keywords: *PATRICIA-trie, OPC-trie, specification of an optimal alphabetical classifier*
DOI: 10.14357/20790279210108

References

1. *Emelyanov N.E., Tishchenko V.A.* Representation of hypertext in the NIKA DBMS // Technology of programming and data storage / Sat. Proceedings of the ISA RAS. T.45. Ed. Corresponding Member RAS Arlazarov V.L. and Doctor of Technical Sciences prof. Emelyanov N.E. - M. 2009. P. 17-36.
2. *Briandais R.* File Searching Using Variable Length Keys / R. Briandais // Proc. AFIPS Western Joint Computer Conference, San Francisco, California, USA, 15 March 1959. P. 295-298.
3. *Godunov A.N.* NIKA DBMS / A.N. Godunov, N.E. Emelyanov, A.N. Kosmynin, V.A. Soldatov // Database and knowledge management systems. M.: "Finance and Statistics", 1991. P.209-249.
4. *Morrison D.* PATRICIA-practical algorithm to retrieve information coded in alphanumeric / D. Morrison // J. ACM 15.4 (0ct. 1968). P 514-534.
5. *Sussenguth E.H.* Use tree structures for processing files / E.H. Sussenguth // CACM 6, 1963, P. 272-279.
6. *Tishchenko V.A.* The choice of the optimal alphabetical classifier while minimizing the total number of operations // Proceedings of ISA RAS, 2018. V. 68. No. 1. P.54-57
7. *Arlazarov V.L.* Device for finding information by keywords / V.L. Arlazarov, V.A. Tishchenko // Patent for invention No. 2679967 C1 Russian Federation, 2019. Bul. No. 5
8. *Tishchenko V.A.* Implementation of a lexicographic classifier for OODBMS NIKA / V.A. Tishchenko // Materials of the XXXIII International Scientific and Practical Conference "Eurasiascience", November 15, 2020. P. 69-71.

Tishchenko Vladimir Alexandrovich. Researcher, ISA FRC CSC RAS. Employee of department of Informatics, PSTGU. Graduated from the MEPhI in 1993. Number of publications: 24. Research interests: means of creation and support of electronic libraries and electronic publications. E-mail: vtischenko@isa.ru