УДК 004.832

СПОСОБЫ РЕАЛИЗАЦИИ НЕЧЕТКОГО ПОИСКА В POSTGRESQL

Байрамова Х.Б. (Университет ИТМО)
Научный руководитель – преподаватель, Николаев В.В.
(Университет ИТМО)

Введение. Нечеткий поиск представляет собой поиск не только по заданному образцу, но и по близким к этому образцу значениям. Информация, полученная в результате данного вида поиска, отображается более полно за счет своей вероятностной природы, что позволяет удовлетворить поисковый запрос пользователя даже при условии частичной некорректности запрашиваемой информации. Нечеткий поиск уже реализован в таких системах , как: ElasticSearch, Microsoft Azure, Sphinx. Но отсутствие согласованности данных, оперирование материализованными документами без возможности индексации представлений, полученных путем соединения документов, отсутствие мгновенной индексации документов - существенные недостатки, из-за которых встает вопрос о реализации нечеткого поиска в PostgreSQL. В СУБД есть модули, позволяющие осуществить нечеткий поиск на начальных этапах. Но и они обладают рядом недостатков: тесная интеграция с латинскими символами, отсутствие сформированного представления, невозможность определения кандидата на исправление при нескольких опечатках. Поэтому имеет смысл рассмотреть проблему реализации нечеткого поиска в PostgreSQL более детально.

Основная часть. Одним из главных вопросов при реализации нечеткого поиска является выбор оптимального алгоритма для конкретной задачи. На данный момент существует множество алгоритмов, осуществляющих нечеткий поиск.

1. Алгоритм Левенштейна (расстояние Левенштейна) - позволяет вычислить минимальное количество односимвольных операций (вставка, удаление, замена) необходимых для превращения одной строки в другую. Недостаток алгоритма заключается в том, что он не учитывает схожесть звуков и не может работать с синонимами, а так же плохо работает со словосочетаниями.

2. Алгоритм Дамерау-Левенштейна - расширение алгоритма Левенштейна, которое включает в себя транспозицию (перестановку) соседних символов, поэтому, в отличие от алгоритма Левенштейна, позволяет задействовать при поиске словосочетания.

3. Алгоритм Фонетического кодирования (Soundex, Metaphone, Double Metaphone) - алгоритмы, которые преобразуют слова в строку символов на основе их звукового произношения. Это позволяет искать слова, которые звучат похоже, но могут быть записаны по-разному. Особенно эффективны при работе с именами собственными.

4. Алгоритмы на основе N-грамм - алгоритмы, которые разбивают слова на последовательности символов фиксированной длины и сравнивают их между собой. Этот подход позволяет учитывать перестановки символов внутри слова. Но при увеличении количества допущенных опечаток, эффективность его уменьшается.

На данный момент алгоритм на основе триграмм и фонетические алгоритмы реализованы в PostgreSQL в модулях pg_trgm и fuzzystrmatch соответственно. Но они имеют ряд недостатков: алгоритм триграмм - свойственную им неэффективность при увеличении количества опечаток, фонетические алгоритмы - реализацию только для латиницы.

Алгоритмы нечеткого поиска имеют свои достоинства и недостатки, но, что самое главное, каждый из алгоритмов эффективен в зависимости от определенных условий. Под условиями подразумеваются принадлежность слов в словаре к именам собственным, количество опечаток в слове, изолирован ли запрашиваемый термин и так далее. Таким образом, имеет смысл исследовать алгоритмы с точки зрения их применимости и выявить случаи, когда тот или иной алгоритм будет наиболее полезен (в разрезе критериев эффективности и осуществления качественного поиска).

Помимо рассмотрения алгоритмов и их направленности, так же исследуются статистические данные, которые могут повысить эффективность использования алгоритмов. Под статистическими данными понимаются параметры и средства, при анализе которых, можно судить о целесообразности выбора того или иного алгоритма. К примеру, указание количества букв при добавлении позиции в словарь или создания индекса для группировки схожих слов.

Выводы. В результате работы в СУБД PostgreSQL добавляются и дорабатываются реализации алгоритмов нечеткого поиска, для облегчения использования вводится оператор.

Список использованных источников:

1. Маннинг К. Д., Рагхаван П., Шютце Х. Введение в информационный поиск. – 2011. – С.69 – 84
2. Бойцов Л. М. Синтез системы автоматической коррекции, индексации и поиска текстовой информации : дис. – –М.: Моск. акад. рынка труда и информационных технологий, 2003.–138 с, 2003.
3. Zobel J., Dart P. Phonetic string matching: Lessons from information retrieval //Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval. – 1996. – С. 166-172.