ИССЛЕДОВАНИЕ ВЕРОЯТНОСТНЫХ АЛГОРИТМОВ И СТРУКТУР ДАННЫХ

Пискунов А.С.
Лицей при ТПУ, г. Томск

Научный руководитель: Буркатовская Ю.Б., к.ф.-м.н., доцент кафедры информационных систем и технологий ТПУ

Актуальность использования вероятностных алгоритмов определяется необходимостью экономного расхода памяти. Они позволяют решать задачи, которые возможно решить другим способом, но слишком дорогим, требующим много времени и других ресурсов, помогают создавать более дешевые и более предсказуемые системы.

Вероятностные алгоритмы (probabilistic algorithm) предназначены для решения задач, точное решение которых является невозможным или нерациональным.

На практике используется достаточно большое количество вероятностных алгоритмов, но я бы хотел остановиться на двух: фильтр Блума и алгоритм HyperLogLog.

Фильтр Блума (англ. Bloom filter) – это вероятностная структура данных, придуманная Бёртоном Блумом в 1970 году, позволяющая проверять принадлежность элемента к множеству. При этом существует возможность получить ложноположительное срабатывание (элемента в множестве нет, но структура данных сообщает, что он есть), но не ложноотрицательное.

Фильтр Блума представляет собой битовый массив из m бит. Изначально, когда структура данных хранит пустое множество, все m бит обнулены. Пользователь должен определить k независимых хеш-функций $h_1(e), \dots, h_k(e)$, отображающих каждый элемент в одну из m позиций битового массива достаточно равномерным образом.

Для добавления элемента e необходимо записать единицы на каждую из позиций $h_1(e), \dots, h_k(e)$ битового массива.

Для проверки принадлежности элемента e к множеству хранимых элементов, необходимо проверить состояние битов $h_1(e), \dots, h_k(e)$. Если хотя бы один из них равен нулю, элемент не может принадлежать множеству (иначе бы при его добавлении все эти биты были установлены). Если все они равны единице, то структура данных сообщает, что *e* принадлежит множеству. При этом может возникнуть две ситуации: либо элемент действительно принадлежит множеству, либо все эти биты оказались установлены по случайности при добавлении других элементов, что и является источником ложных срабатываний в этой структуре данных.

Фильтр Блума актуально используется в прокси-серверах для опции cache digests; в Google BigTable для уменьшения числа обращений к

жесткому диску при проверке на существование заданной строки или столбца в таблице базы данных; в компьютерных программах для проверки орфографии.

При этом достоинства описываемой структуры данных по сравнению с хеш-таблицами, является возможность обходиться на несколько порядков меньшими объёмами памяти, пренебрегая погрешностями. Обычно алгоритм используется для уменьшения числа запросов к несуществующим данным в структуре данных с более дорогостоящим доступом (например, расположенной на жестком диске или в сетевой базе данных), то есть для «фильтрации» запросов к ней.

HyperLogLog – это алгоритм, используемый для подсчета уникальных элементов во множестве.

HyperLogLog – это расширение более раннего алгоритма LogLog, являющегося результатом алгоритма 1984 Флайолет-Мартин. Так как для вычисления точной мощности мультимножества требуется объем памяти, пропорциональный мощности, использование наивных алгоритмов становится нецелесообразно, потому что затрачивается большой объем памяти, в этих случаях оптимально использовать описываемую структуру данных. Алгоритм HyperLogLog, использует значительно меньше памяти, но ценой получения только приближения мощности. Алгоритм HyperLogLog способен оценивать мощности $>10^9$ с типичной точностью 2%, используя 1,5 кбайт памяти.

При изучении вероятностных алгоритмов становится очевидно, что использование таких алгоритмов не только приемлемо, более того, имеет смысл специально искать возможность применить их. Несмотря на присутствие погрешности, вероятностные алгоритмы целесообразно использовать на больших данных. В информационных источниках мы встречаем опасение по поводу использования вероятностных алгоритмов, предположения о том, что они могут работать неэффективно или их невозможно отладить. Однако эти опасения беспочвенны. Вероятностные алгоритмы содержат в себе долю случайности, которая поддается количественной оценке. Эту случайность можно проанализировать и получить надежный прогноз относительно поведения алгоритма. На самом деле, доля неопределенности в поведении алгоритма очень мала.

Список информационных источников

1. Дасгупта С., Пападимитриу Х., Вазирани У. Алгоритмы. – М.: МЦНМО, 2014. – 320 с.

2. Bloomfilter [Электронный ресурс]. – режим доступа: https://en.wikipedia.org/wiki/Bloom_filter. 10.09.2017.

3. HyperLogLog [Электронный ресурс]. – режим доступа: https://en.wikipedia.org/wiki/HyperLogLog. 10.09.2017.