АВТОМАТИЗИРОВАННЫЙ АНАЛИЗ ИЗМЕНЕНИЙ ЗНАЧЕНИЙ ПАРАМЕТРОВ НА БОЛЬШИХ СОВОКУПНОСТЯХ ДАННЫХ

С.И. Мыц

Разработчик в компании Яндекс (ООО «ЯндексБел»), ассистент кафедры информатики факультета компьютерных систем и сетей БГУИР, Республика Беларусь

Белорусский Государственный Университет Информатики и Радиоэлектроники, sergei.myts@gmail.com

We describe an instrument and an approach it is based on which allows to automatize analysis of a large-scale system parameter changes. The method is based on the idea of utilizing data slices. They are used to determine the subsets that are most influential to the selected value changes. This technique is used in the instrument that makes it possible to perform relatively fast analysis of required parameter changes at the given time frame.

При работе крупных систем генерируются большие объёмы данных. Если возникает потребность оптимизировать использование этих систем, встаёт задача определять важные изменения в этих данных. В условиях, когда этих данных много, появляется необходимость каким-то образом понимать, когда в них возникают аномалии или проверять правильность интерпретации гипотез о них. Для этого введём специальные параметры, которые измеряются по данным, и уже далее будем производить анализ этих величин. При этом возникают ситуации, когда параметры меняются и нужно понять причину этих изменений.

Ниже мы описываем обобщённый подход и представляем его частную реализацию, которая используется в основе инструмента автоматизирующего анализ этих значений и позволяющего аналитикам быстрее находить причины изменений. Подход развивает идеи, упомянутые в магистерской диссертации по теме “Методы сравнения групп запросов по различию в качестве в онлайн экспериментах” [1]. На основе информации этого источника мы выделили основу интересующего нас метода, рассмотрели подробнее его свойства и вариации, а выбрали и создали наиболее подходящую, по нашему мнению, для практического применения реализацию. В качестве примера нами используется поисковая система. Пользователь обращается к ней, вводя запрос, и получает в качестве ответа страницу с поисковой выдачей – списком найденных результатов, отранжированных по убыванию предполагаемой системой релевантности. После этого пользователь взаимодействует со страницей. Для исследования различных аспектов выдачи и свойств поведения

пользователя можно исследовать информацию об этом взамодействии: запросы к системе, клики по элементам поисковой выдачи (нажатие на активные элементы и переходы по веб-ресурсам) и их характеристики [2][3].

Для упрощения рассмотрения такой информации введём некоторые числовые величины, которые отражают интересные нам свойства. Такие величины принято называть метриками [3]. Мы рассматриваем метрику как осмысленную агрегированную величину, которая рассчитывается по некоторому множеству запросов и связанных с ними действий.

Примеры метрик [1][3]:

1. Среднее количество кликов в рамках запроса.
2. Доля некликнутых запросов.
3. Количество запросов на одну пользовательскую сессию.
4. Время до первого клика.
5. CTR позиции выдачи – click-through rate – отношение количества кликов на результате к количеству его показов.

На множестве запросов можно ввести функцию-предикат (со значениями ноль-один), с помощью которой задаётся срез данных – подмножество запросов – те из них, для которых предикат даёт единицу. Такие предикаты необходимо выбирать в зависимости от специфики системы и преследуемых целей.

Примеры срезов множества запросов к поисковой системе [1][3]:

1. Тип запроса: навигационный, коммерческий и т.д.
2. Частотность: частый или редкий.
3. Регион пользователя.
4. Используемая формула ранжирования.
5. Другие признаки.

Задав основные определения, перейдём к общему описанию предлагаемого механизма поиска причин изменений. В случае значительного изменения некоторой метрики возникает необходимость понимания причин и механизмов, приведших к этому изменению. Для осуществления анализа необходимо иметь достаточно большое и представительное, в смысле отражения разных аспектов внутренней структуры данных, множество срезов. После этого перейдём к ранжированию срезов по убыванию их “влияния” на изменение метрики на всём множестве данных. В результате рассмотрения срезов, оказавшихся в числе первых, можно провести их интерпретацию и сделать выводы о возможных причинах изменений. Вопрос выбора механизма поиска таких срезов нетривиален. Ввиду того, что понятие «влияния» не определено чётко, ранжировать срезы можно разными способами, исходя из того, с какой точки зрения рассматривается данная задача. Пример применения с такой целью анализа чувствительности, линейной регрессии и коэффициентов Соболя можно найти в уже упомянутой работе [1]. Кроме этого, можно

выбирать способ из множества различных вариантов. В рамках данного текста мы рассматриваем один конкретный и, как утверждается, более практичный и простой в реализации подход. Рассмотрим одно из семейств алгоритмов машинного обучения – деревья решений. В результате работы некоторых алгоритмов из этого семейства, в частности, например, в алгоритме random forest, побочным эффектом можно получить величины, характеризующие полезность отдельных входных параметров в предсказании значения целевой функции [4][5][6]. Построим функцию, аргументами которой будут ноль-один значения принадлежности запроса к срезу, а значением – значение метрики на таком запросе. Теперь, с помощью деревьев решений, построим предсказатель значения метрики на совокупности срезов. В результате этого мы получаем полезность отдельных срезов для предсказания значения метрики. Теперь будем ранжировать срезы по убыванию этой полезности, взяв её в качестве “влияния”, и анализировать полученные результаты ранжирования срезов. Для создания инструмента реализующего описанный выше подход выполнено следующее:

1. Собрано достаточно представительное множество срезов.

2. Создан механизм, который будет делать необходимые выжимки данных по множеству запросов и значению метрики на совокупностях срезов. Для этого использовались инструменты массовых вычислений под парадигмой MapReduce.

3. Написана программа, которая по выжимкам строит функцию и выдаёт отранжированные срезы.

4. Предоставлен пользовательский интерфейс, позволяющий выполнять запросы к такому инструменту и просматривать результаты.

5. Подход, описанный выше, позволил автоматизировать анализ и помогает в исследовании изменений большой системы. Благодаря этому можно оперативнее реагировать на проблемы, а также лучше и точнее анализировать результаты вносимых в систему правок.

Литература

1. Валгушев Д.Н., Методы сравнения групп запросов по различию в качестве в онлайн экспериментах // Москва, 2015

2. Manning C. D., Raghavan. P, Schütze H., Introduction to Information Retrieval // Cambridge University Press, 2009

3. Chapelle O., Joachims T., Radlinski F., Yue Y., Large-Scale Validation and Analysis of Interleaved Search Evaluation // ACM Transactions on Information Systems (TOIS), 2012

4. Brieman L., Random forests // University of California, 2001

5. Sandri M., Zuccolott P., A bias correction algorithm for the Gini variable importance measure in classification trees // University of Brescia - Department of Quantitative Methods, 2008

6. Breiman L, Cutler A., Liaw A., Wiener M., Breiman and Cutlers Random Forests for Classification and Regression. R package version 4.6-10 // CRAN, 2014