УНИВЕРСИТЕТ ИТМО
НАУЧНО-ТЕХНИЧЕСКИЙ ВЕСТНИК ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ
май–июнь 2017 Том 17 № 3 ISSN 2226-1494 http://ntv.ifmo.ru/
SCIENTIFIC AND TECHNICAL JOURNAL OF INFORMATION TECHNOLOGIES, MECHANICS AND OPTICS
May–June 2017 Vol. 17 No 3 ISSN 2226-1494 http://ntv.ifmo.ru/en

УДК 004.056.55

ОЦЕНКА ПОДОБИЯ ДЕРЕВЬЕВ С ПОМОЩЬЮ ВЫЧИСЛЕНИЯ *pq*-ГРАММ РАССТОЯНИЯ

А.Г. Андреева[a], Т.А. Маркина[a]

[a] Университет ИТМО, Санкт-Петербург, 197101, Российская Федерация
Адрес для переписки: Tiari@mail.ru

Информация о статье
Поступила в редакцию 01.03.17, принята к печати 12.04.17
doi: 10.17586/2226-1494-2017-17-3-490-497
Язык статьи – русский

Ссылка для цитирования: Андреева А.Г., Маркина Т.А. Оценка подобия деревьев с помощью вычисления *pq*-грамм расстояния // Научно-технический вестник информационных технологий, механики и оптики. 2017. Т. 17. № 3. С. 490–497. doi: 10.17586/2226-1494-2017-17-3-490-497

Аннотация
Представлен алгоритм оценки подобия иерархических данных на основе вычисления *pq*-грамм расстояния. Выполнен анализ чувствительности алгоритма от выбранных параметров *p* и *q*. Показано, насколько сильно будет изменяться результат работы алгоритма при сравнении двух деревьев, имеющих различие в одном произвольном узле, когда один из узлов исходного дерева удален, переименован, либо добавлен лишний узел. Продемонстрировано, что подобный анализ позволяет подобрать параметры *p* и *q* применительно к решаемой задаче. Обоснована задача предварительной оценки дерева – приближенный анализ начального уровня расхождений узлов в выбранных *pq*-граммах сравниваемых деревьев. Обозначены основные термины и определения, относящиеся к алгоритмам обработки древовидных структур данных, а также непосредственно к самому рассматриваемому алгоритму. Приведены примеры, иллюстрирующие практическое использование алгоритма, показаны детали реализации алгоритма на реальной задаче.

Ключевые слова
pq-грамм, деревья, подобие деревьев, *pq*-грамм расстояние, *pq*-грамм шаблон, *pq*-грамм профиль, XML

TREE SIMILARITY ESTIMATION BY CALCULATION OF *pq*-GRAM DISTANCE

A.G. Andreeva[a], T.A. Markina[a]

[a] ITMO University, Saint Petersburg, 197101, Russian Federation
Corresponding author: Tiari@mail.ru

Article info
Received 01.03.17, accepted 12.04.17
doi: 10.17586/2226-1494-2017-17-3-490-497
Article in Russian
For citation: Andreeva A.G., Markina T.A. Tree similarity estimation by calculation of *pq*-gram distance. *Scientific and Technical Journal of Information Technologies, Mechanics and Optics*, 2017, vol. 17, no. 3, pp. 490–497 (in Russian). doi: 10.17586/2226-1494-2017-17-3-490-497

Abstract
The paper presents an algorithm for similarity estimation of hierarchical data based on the *pq*-gram distance calculation. The dependence of the algorithm sensitivity on the selected parameters *p* and *q* is analyzed. We show how much the result of the algorithm will change at comparing of two trees that have difference in one random node when one of the nodes of the source tree is deleted, renamed, or an extra node is added. It is demonstrated that such analysis enables to select the parameters *p* and *q* in relation with the solving problem. The problem of a tree preliminary evaluation is substantiated - an approximate analysis of the initial level of node differences in the selected *pq*-grams of the compared trees. The basic terms and definitions relating to the tree-based data structuring algorithms are described. Examples of the algorithm practical application and the details of its implementation on a real problem are shown.

Keywords
pq-gram, trees, tree similarity, *pq*-gram distance, *pq*-gram pattern, *pq*-gram profile, XML

Введение

Вычисление *pq*-грамм-расстояния (*pq*-gram distance) позволяет определить меру сходства двух иерархических объектов. В общем случае задача сравнения больших объемов данных с древовидной структурой представляется сложной и ресурсоемкой, особенно если данные имеют различное представление и не содержат общих ключей. Часто в качестве метрики выбирается расстояние редактирования [1] – это подсчет минимального количества операций вставки, удаления и переименования узла, необходимых для преобразования одного дерева в другое. Например, расстояние редактирования по алгоритму Zhang and Sasha [2] вычисляется за время $O(n^2 \log^2 n)$ и требует $O(n^2)$ памяти для деревьев с глубиной $O(\log n)$, где n – число узлов. При этом в худшем случае (в зависимости от формы дерева) время увеличивается до $O(n^4)$. Алгоритмы Demaine [3] и RTED [4] требуют $O(n^2)$ памяти и работают за максимальное время $O(n^3)$. Напротив, вычисление *pq*-грамм расстояния занимает $O(n \log n)$ времени и задействует $O(n)$ памяти, позволяя быстро выполнить примерную оценку подобия структур данных большого объема с учетом не только значений узлов, но и связей между ними.

pq-грамм-расстояние позволяет выполнить приблизительную оценку подобия деревьев за меньшее время, чем другие, более точные алгоритмы сравнения деревьев за счет того, что оперирует при расчетах не узлами, а группами узлов – поддеревьями, а также за счет универсального подхода к вычислению функции расстояния.

Основные сведения об алгоритме

Алгоритм вычисления *pq*-грамм был представлен в 2005 г. на 31-й конференции VLDB [5]. В общем виде задача, решаемая с использованием алгоритма вычисления *pq*-грамм-расстояния, представляется следующим образом: у компании имеется несколько офисов, каждый из которых содержит свою независимую базу данных с адресами квартир и фамилиями владельцев, а центральный офис решил объединить эти данные в единую базу [5]. Изначальная база данных представлена двумя таблицами: первая – название улицы и соответствующий ей идентификатор; вторая связывает идентификатор и адрес квартиры в виде: номер дома, номер корпуса, номер квартиры. Проблемы объединения заключаются в том, что имена улиц могут быть введены с ошибками или иметь различный формат, в результате чего строки с именем получаются неидентичными в разных таблицах; идентификатор улицы в одной базе данных не совпадает с идентификатором той же улицы в другой; а также в одном из офисов могут быть не полные сведения обо всех квартирах, находящихся на одной улице. Для решения такой проблемы можно отображать конечный адрес в виде дерева, в котором корнем выступает название улицы, а соответствующие ей адреса – дочерними узлами. На рис. 1 показано графическое представление задачи.

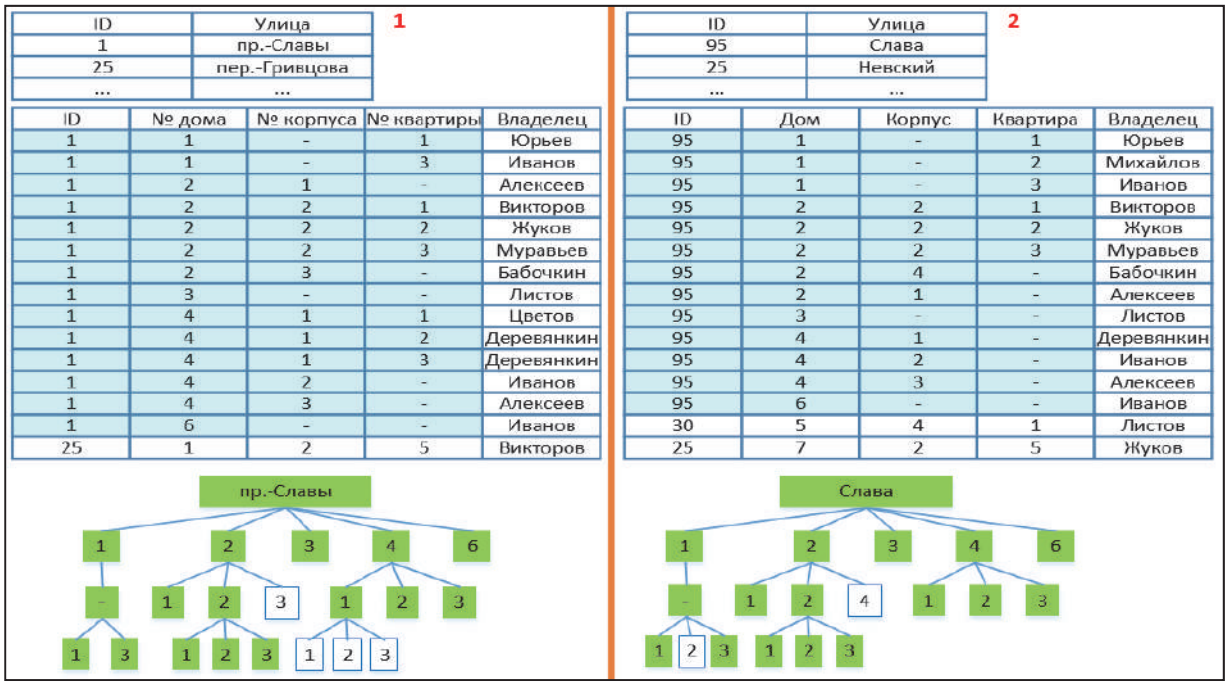

Рис. 1. Представление адресов из базы данных в виде дерева

Из рис. 1 видно, что вместо того, чтобы выискивать похожие имена улиц в различных базах данных, можно сравнивать адреса (поддеревья) и при большом числе совпадений уже работать с конкретной улицей,

объединяя адреса из разных баз. Отличающиеся же деревья можно считать различными улицами и добавлять их в общую базу под отдельными записями. При таком подходе не требуется детально сравнивать каждую пару узлов на высоте *h*, как в случае с расстоянием редактирования, а лишь оценить их схожесть dist $(T1(ID), T2(ID)) \le \tau$, где τ – это выбранный порог, разделяющий похожие и не похожие деревья.

Пусть дерево T – это связный направленный ациклический граф $T = (V, E)$ с узлами $V(T) = V$ и ребрами $E(T) = E$. Ребро – это направленная пара (p, c), причем $p, c \in V(T)$ – узлы, и узел p является родителем для узла c. Узлы с общим родителем – дочерние (или родственные) узлы. Между дочерними узлами может быть определен порядок «≤». Дочерние узлы s_1 и s_2 являются смежными, если между ними нет другого родственного узла x ($s_1 \ne x \ne s_2$) такого, что $s_1 \le x \le s_2$. Потомками узла p называются все его дочерние узлы и далее по цепочке. Узел d называется i-м потомком узла p, если от узла p до d включительно находится i узлов-потомков для p. Степенью узла p называется число его дочерних узлов (ближайших потомков). Узел, не имеющий родителя – корень дерева. Узел, не имеющий потомков – лист. Каждый узел a, стоящий на пути от корня дерева до произвольного узла v, называется предком узла v. Если путь от узла a до v занимает длину из k узлов, то узел a называется предком узла v на расстоянии k. Родитель узла является предком на расстоянии 1. Уровнем (высотой) узла называется длина пути (число узлов) от корня до этого узла. Глубиной дерева T называется расстояние от корня дерева до любого самого удаленного листа дерева (самый длинный путь в дереве).

Метка узла (значение) – это символ $\varepsilon \in \Sigma$, где Σ – это конечный алфавит. Каждый узел дерева T $(v \in V(T))$ имеет собственную метку $l(v)$. Узел o, имеющий специальную метку $l(o) = *$, называется пустым узлом. Далее каждый узел будет представлен парой (идентификатор узла, метка). Дочерние узлы имеют последовательность обхода слева направо. Подробнее о древовидных структурах данных и методах обхода узлов можно посмотреть в [6–9].

На рис. 2 представлено дерево $T_1 = (V, E)$, причем $V = \{v_1, v_2, v_3, v_4, v_5, v_6\}$, $E = \{(v_1, v_2), (v_1, v_5), (v_1, v_6), (v_2, v_3), (v_2, v_4)\}$, последовательность узлов $v_2 \le v_5 \le v_6, v_3 < v_4$. Корневым узлом дерева T_1 является узел v_1 – $\mathrm{root}(T) = v_1$ – он является предком для всех остальных узлов дерева. У узла v_1 три прямых потомка (дочерних узла): родственные узлы – v_2, v_5 и v_6. Метками узлов дерева T_1 являются $l(v_1) = a, l(v_2) = a, l(v_3) = e, l(v_4) = b, l(v_5) = b, l(v_6) = c$, что показано на рис. 2, при этом v и w – это идентификаторы узлов.

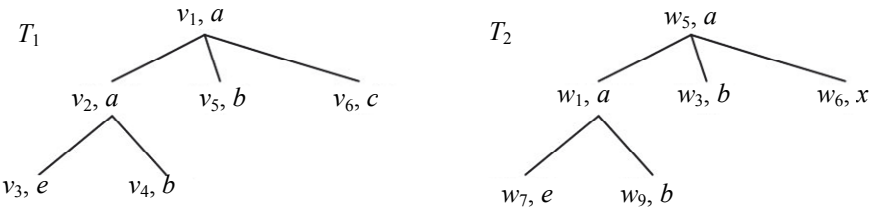

Рис. 2. Сравниваемые деревья T_1 и T_2, узлы записаны как: идентификатор, значение (метка)

Поддеревом $S \in T$ называется дерево, узлы которого являются подмножеством множества узлов дерева T: $V(S) \subseteq V(T)$ и $E(S) \subseteq E(T)$, с сохранением первоначальной последовательности узлов и их отношений. При обходе дерева сначала посещается корень дерева, затем рекурсивно обходятся все поддеревья с корнями в его дочерних узлах. i-м узлом дерева T называется такой узел v, который посещают на i-ой итерации обхода дерева (i-й посещаемый узел).

Два дерева называют изоморфными, если существует взаимно однозначное соответствие (биекция) между совокупностями их вершин и ребер [10].

Например, объединяя все вышесказанное и смотря на рис. 2, можно видеть, что дерево $S_1 = (\{v_2, v_3, v_4\}, \{(v_2, v_3), (v_2, v_4)\}), v_3 \le v_4$ – это поддерево дерева T_1. При обходе дерева T_1 узлы должны посещаться в следующем порядке: $v_1, v_2, v_3, v_4, v_5, v_6$. Деревья T_1 и T_2 изоморфны, так как существует биекция $m = \{(v_1, w_5), (v_2, w_1), (v_3, w_7), (v_4, w_9), (v_5, w_3), (v_6, w_6)\}$.

Если рассматривать в общем, то *pq*-граммы – это все поддеревья рассматриваемого дерева определенной формы. Чтобы в процессе работы алгоритма все узлы рассматриваемого дерева присутствовали хотя бы в одной *pq*-грамме, оно дополняется пустыми узлами, тогда *pq*-граммы – это все поддеревья определенной формы такого дополненного дерева.

pq-расширенное дерево. Пусть дано дерево T и два числа $p > 0$ и $q > 0$, тогда *pq*-расширенное дерево T^{pq} строится из дерева T путем добавления к нему ($p-1$) предка к корню, вставкой ($q-1$) до-

черних узлов перед первым дочерним узлом и после последнего дочернего узла исходного дерева к каждому нелистовому узлу дерева (перед группой дочерних для него родственных узлов и после), а также добавляется *q* детей к каждому листовому узлу дерева *T*. Все добавленные узлы являются пустыми и не должны встречаться в изначальном дереве *T*. На рис. 3 показано *pq*-расширенное дерево $T_1^{2,3}: p=2, q=3$, где o_i, l – идентификатор узла (*o*), метка узла (*l*).

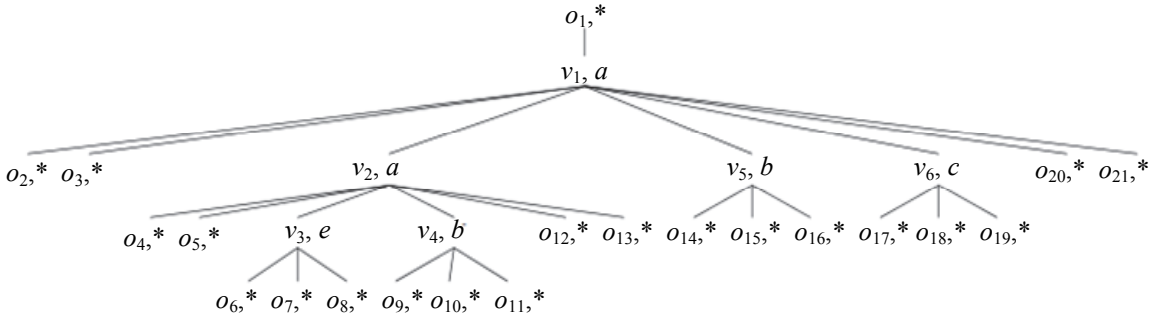

Рис. 3. *pq*-расширенное дерево $T_1^{2,3}$ дерева T_1 с рис. 2

pq-грамм-шаблон: для $p>0$ и $q>0$ *pq*-грамм-шаблон – это дерево, которое содержит корневой узел с добавленным к нему ($p-1$) предком и *q* дочерними узлами. Например, 2,3-грамм шаблон – это дерево $(\{p_1,p_2,p_3,p_4,p_5\},\{(p_1,p_2),(p_2,p_3),(p_2,p_4),(p_2,p_5)\}), p_3 \le p_4 \le p_5$. p_2 – это корень дерева, который имеет одного ($p-1$) предка (p_1) и 3 (*q*) дочерних узла p_3, p_4, p_5.

pq-грамм: для $p>0$ и $q>0$ *pq*-грамм *G* дерева *T* – это поддерево расширенного дерева T^{pq} со следующими свойствами: поддерево *G* изоморфно *pq*-грамм шаблону, и смежные родственные узлы в *G* также являются смежными родственными узлами в T^{pq}.

Кортеж меток: пусть *G* – это *pq*-грамма с узлами $V(G)=\{v_1,...,v_p,v_{p+1},...,v_{p+q}\}$, где v_i – это *i*-й узел в последовательности обхода *pq*-граммы, тогда кортеж меток *pq*-граммы *G* равен $l(G)=(l(v_1),...,l(v_p),l(v_{p+1}),...,l(v_{p+q}))$. Название *pq*-грамма может быть применимо не только непосредственно к *pq*-грамме, но и к кортежу меток. Иногда кортеж меток также называют *pq*-грамм индексом. На рис. 4 приведены некоторые 2,3-граммы дерева T_1.

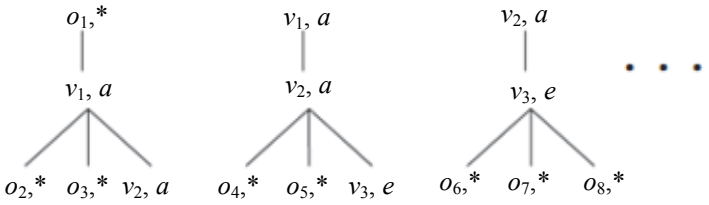

Рис. 4. Частичный набор 2,3-грамм дерева T_1

2,3-граммы с рис. 4 получены путем передвижения 2,3-грамм-шаблона по расширенному дереву $T_1^{2,3}$ с рис. 3. Шаблон совмещается верхушкой дерева, т.е. для первой *pq*-граммы корнем является узел v_1, а дочерними узлами являются два пустых узла и узел v_2. Соответствующий этой 2,3-грамме кортеж меток: (*,*a*,*,*,*a*) – обход всегда слева направо.

pq-грамм-профиль: для $p>0$ и $q>0$ *pq*-грамм-профиль $P^{p,q}(T)$ дерева *T* – это полный набор кортежей меток $l(G_i)$ для всех *pq*-грамм G_i дерева *T*. На рис. 5 показаны *pq*-грамм профили деревьев T_1 и T_2 с рис. 2. *pq*-граммы в профиле могут повторяться, если одинаковые кортежи меток в расширенном дереве встречаются несколько раз.

pq-грамм-расстояние показывает, сколько общих *pq*-грамм содержат *pq*-грамм-профили сравниваемых деревьев. *pq*-грамм-расстояние: для $p>0$ и $q>0$ *pq*-грамм расстояние $\Delta^{p,q}(T_1,T_2)$ между деревьями T_1 и T_2 определяется формулой (1).

$$\Delta^{p,q}(T_1,T_2)=1-2\frac{\left|p^{p,q}(T_1)\cap p^{p,q}(T_2)\right|}{\left|p^{p,q}(T_1)\cup p^{p,q}(T_2)\right|}. \quad (1)$$

Например, для деревьев T_1 и T_2 с рис. 2 *pq*-грамм расстояние будет вычисляться следующим образом: $\left|p^{p,q}(T_1)\cap p^{p,q}(T_2)\right|=9:\{(*;a;*;*;a);(a;a;*;*;e);(a;e;*;*;*);(a;a;*;e;b);(a;b;*;*;*);(a;a;e;b;*);$

$(a;a;b;*;*);(*;a;*;a;b);(a;b;*;*;*)\}$. $\left|p^{p,q}(T_1)\cup p^{p,q}(T_2)\right|=26$ – число всех pq-грамм для деревьев T_1 и T_2. Тогда pq-грамм-расстояние между T_1 и T_2 будет равно

$$\Delta^{2,3}(T_1,T_2)=1-2\frac{9}{26}=0{,}31.$$

$P^{2,3}(T_1)$ профиль	$P^{2,3}(T_2)$ профиль
(*, a, *, *, a)	(*, a, *, *, a)
(a, a, *, *, e)	(a, a, *, *, e)
(a, e, *, *, *)	(a, e, *, *, *)
(a, a, *, e, b)	(a, a, *, e, b)
(a, b, *, *, *)	(a, b, *, *, *)
(a, a, e, b, *)	(a, a, e, b, *)
(a, a, b, *, *)	(a, a, b, *, *)
(*, a, *, a, b)	(*, a, *, a, b)
(a, b, *, *, *)	(a, b, *, *, *)
(*, a, a, b, c)	(*, a, a, b, x)
(a, c, *, *, *)	(a, x, *, *, *)
(*, a, b, c, *)	(*, a, b, x, *)
(*, a, c, *, *)	(*, a, x, *, *)

Рис. 5. 2,3-грамм-профили деревьев T_1 и T_2

Размер pq-грамм профиля для дерева с l листьями и i внутренними (не листовыми) узлами вычисляется по формуле (2).

$$\left|\Delta^{p,q}(T_1,T_2)\right|=2l+qi-1. \quad (2)$$

pq-грамм расстояние равно 1, если деревья не содержат общих pq-грамм, и 0, если их pq-грамм профили идентичны. Необходимо помнить, что нулевое расстояние не означает совпадение деревьев – они могут быть зеркальным отражением друг друга. Простейший алгоритм создания pq-грамм-профиля и вычисления расстояния приведен в [5].

Расчет зависимости *pq*-грамм расстояния от параметров *p* и *q*

Рассмотрим зависимость значения pq-грамм расстояния от выбранных параметров p и q.

Пусть T_1 – это рассматриваемое дерево, а T_2 – это дерево, полученное из T_1 после одной из следующих операций: переименование одного произвольного узла (замена метки), вставка одного нового узла в произвольное место, удаление одного любого узла. Задача: найти зависимость значения pq-грамм-расстояния между деревьями T_1 и T_2 от выбранных параметров p и q. Решение данной задачи позволит оптимально выбирать значения p и q при использовании алгоритма.

Пусть дерево T_1 высотой H формируется с помощью следующего алгоритма: на каждой итерации обрабатывается один узел и, начиная с корневого узла, добавляются дочерние узлы. Каждый узел с вероятностью P_0 имеет дочерние узлы – значения ИСТИНА (обладает) или ЛОЖЬ (нет дочерних узлов) получаются с помощью биномиального распределения $B(P_0)$. Число дочерних узлов вычисляется через равномерное распределение $U[1...N]$. Метки присваиваются узлам путем случайного выбора метки из набора меток Σ. Таким образом можно получить дерево, форма которого будет приближена к реальным данным.

Пусть n_k – это число узлов в дереве, l_k – число листьев, m_k – число нелистовых (внутренних) узлов на высоте h ($h \le H$), где H – максимальная высота дерева (высота дерева T_1), тогда по формулам (3)–(5) можно рассчитать среднее число узлов на уровне h:

$$E(n_k)=\left(P_0\frac{N+1}{2}\right)^h=a^h. \quad (3)$$

$$E(l_k)=\begin{cases}E|n_k|(1-P_0) & h<H\\ E|n_k| & h=H\end{cases}. \quad (4)$$

$$E(m_k)=\begin{cases}E|n_k|(P_0) & h<H\\ 0 & h=H\end{cases}. \quad (5)$$

Здесь $E(x)$ – математическое ожидание [11] – среднее число узлов на уровне дерева h. Вывод формул представлен в работе [12]. Для вычисления pq-грамм-расстояния применяется нормализованная формула (6) ($I_1=P^{p,q}(T_1)$):

$$\Delta^{p,q}{}_{norm} = \frac{\Delta^{p,q}(T_1, T_2)}{|I_1 \cup I_2| - |I_1 \cap I_2|}. \quad (6)$$

Деревья T_1 и T_2 отличаются на $(q+1)$ pq-грамм. Следовательно, $\Delta^{p,q}(T_1, T_2) = 2(q+1)$. Тогда формулу (6) можно преобразовать в формулу (7):

$$\Delta^{p,q}{}_{norm} = \frac{2(q+1)}{\Delta^{H-1}(2 - P_0(2-q)) + 2a^H + q}. \quad (7)$$

Вывод формулы (7) и всех последующих представлен в [12].

В случае смены метки узла видна следующая зависимость. Пусть узел C дерева T_1^H (H – высота дерева) переименован (произошла смена метки узла) и получилось дерево T_2^H, тогда формулы (8)–(10) показывают ожидаемое значение pq-грамм-расстояния между деревьями T_1 и T_2 при произвольных параметрах p и q при условии, что дерево T_1 было сформировано по вышеописанному алгоритму. Результаты зависят от положения узла C в деревьях T_1 и T_2:

1. C – листовой узел:

$$\Delta^{p,q}{}_{norm}(T_1, T_2) = \frac{2(q+1)}{a^{H-1}(qP_0 + 2 - P_0 + P_0N)}. \quad (8)$$

2. Узел C находится на высоте h, такой, что $h + p < H$:

$$\Delta^{p,q}{}_{norm}(T_1, T_2) = \frac{2q + P_0(N-1)}{q(P_0a^{H-p} + 1) - P_0(N-1)a^{H-p}}. \quad (9)$$

3. Узел C находится на высоте h, такой, что $h + p > H$:

$$\Delta^{p,q}{}_{norm}(T_1, T_2) = \frac{2q + P_0(N-1)}{q(P_0 + a^{-h}) + 2 - P_0 + 2P_0N}. \quad (10)$$

Анализируя результат, видим, что при переименовании листового узла (8) параметр p не влияет на результат вычисления расстояния, таким образом, в реальных задачах, где деревья имеют большое число изменяемых листовых узлов, можно не акцентировать внимание на параметре p, основное влияние оказывает только параметр q. В деревьях с большим количеством нелистовых узлов (формула (9)) зависимость от параметра p является экспоненциальной, в то время как от параметра q – обратно-линейной. Исходя из этого, зависимость от p больше, чем от q в случае, если изменения происходят в нелистовых узлах, близких к дальним листьям дерева, и выбранный pq-грамм-шаблон не задевает корень исходного дерева при сопоставлении с изменяемым узлом. В формуле (10) зависимость от p пропадает, так как изменения происходят близко к корню дерева, и различными оказываются лишь небольшое число кортежей.

Такие вычисления необходимы при использовании алгоритма, так как, например, если брать задачу с рис. 1, то от выбранных параметров p и q будет зависеть уровень сходства деревьев с адресами квартир для заданной улицы. Необходимо оценить, какие узлы будут иметь максимальные различия: номера квартир (листовые значения), корпуса (дальние внутренние узлы) или номера домов (узлы, ближайшие к корню).

Операция вставки нового листового узла C меньше влияет на pq-грамм-профиль, чем вставка внутреннего (нелистового) узла:

1. Вставка листа l в дерево T_1 для получения T_2:

$$\Delta^{p,q}{}_{norm}(T_1, T_2) = \frac{4q}{a^{H-1}(qP_0 + 2 - P_0 + P_0N)}. \quad (11)$$

2. Вставка нелистового узла на высоте h, такой, что $h + p < H$:

$$\Delta^{p,q}{}_{norm}(T_1, T_2) = \frac{2q + P_0(N-1)}{q(P_0a^{H-p} + 1) + 2a^{H-p}(1 - P_0N)}. \quad (12)$$

3. Вставка нелистового узла на высоте h, такой, что $h + p > H$:

$$\Delta^{p,q}{}_{norm}(T_1, T_2) = \frac{2q + P_0(N-1)}{q(P_0 + a^{-h}) + P_0(N+1)}. \quad (13)$$

Влияние операции вставки узла на pq-грамм расстояние похоже на операцию переименования узла. Формулы (11) и (13) показывают, что параметр p не влияет на значение расстояния. В формуле (12) зависимость видна в участке qP_0a^{H-p}, который присутствовал и в операции переименования, следовательно, зависимость от p и q такая же, как и для операции смены метки. Различия – только в использовании параметров N и P_0.

Операция удаления аналогична операции вставки, так как если после удаления узла из T_1 получается дерево T_2 по условию задачи, то обратно – при добавлении узла к дереву T_2 получится T_1. Так как *pq*-грамм-расстояние является симметричным, $\Delta^{p,q}(T_1,T_2)=\Delta^{p,q}(T_2,T_1)$, то результаты аналогичны результатам операции вставки узла.

Обобщая все вышеизложенные выводы, можно прийти к заключению, что предварительная оценка данных перед использованием алгоритма в большинстве случаев не представляет сложности – необходимо лишь примерно оценить, на каком удалении от корня дерева будет больше всего различий и в зависимости от этого выбрать параметры *p* и *q*. Графики зависимостей для различных операций приведены в [12].

Заключение

Вычисление *pq*-грамм-расстояния позволяет выполнить оценку подобия деревьев за время $O(n\log n)$, что работает быстрее, чем известные на сегодняшний день алгоритмы сравнения.

Не все задачи требуют точного результата сравнения всех наборов иерархических данных. Например, при поиске похожих страниц в формате XML не требуется точное знание их различий [13]. Полученную приблизительную выборку может оценить сам пользователь или уже более точный алгоритм, на вход которому будет подан существенно меньший объем данных.

Алгоритм может применяться для решения следующих задач:

- оценка подобия тематического содержания текстов на основе сравнения формализованной выборки наиболее информативных слов и словосочетаний (формализация описана в [14]);
- сравнение данных в формате XML [15] или близком к нему (например, работа с базой Интернет-ресурсов – простой и быстрый поиск похожих результатов);
- поиск повторяющихся путей при маршрутизации сообщений;
- нечеткий поиск в тексте и словаре, когда при совпадении части *n*-грамм (сочетание букв) слова можно возвращать результат;
- обнаружение плагиата в исходных кодах программ на языках высокого уровня;
- объединение нескольких баз данных в одну с выборкой схожих результатов либо удалением повторяющихся записей.

Порой алгоритм применяется лишь частично – исходное дерево разбивается на *n*-граммы по аналогии с *pq*-граммами, но без вставки пустых узлов. Затем ведется подсчет некоторого набора параметров в зависимости от значений полученных *n*-грамм, после чего можно обрабатывать данные с применением других алгоритмов [16] в зависимости от поставленной задачи.

Литература

1. Tai K.-C. The tree-to-tree correction problem // Journal of ACM. 1979. V. 26. N 3. P. 422–433. doi: 10.1145/322139.322143
2. Zhang K., Shasha D. Simple fast algorithms for the editing distance between trees and related problems // SIAM Journal of Computing. 1989. V. 18. N 6. P. 1245–1262.
3. Demaine E.D., Mozes S., Rossman B., Weimann O. An optimal decomposition algorithm for tree edit distance // ACM Transactions on Algorithms. 2009. V. 6. N 1. Art. 2. doi: 10.1145/1644015.1644017
4. Pawlik M., Augsten N. RTED: a robust algorithm for the tree edit distance // Proc. 38th VLDB Endowment. Istanbul, Turkey, 2012. V. 5. N 4. P. 334–345.
5. Augsten N., Boehlen M., Gamper J. Approximate matching of hierarchical data using pq-grams // Proc. 31st Int. Conf. on Very Large Data Bases. Trondheim, Norway, 2005. V. 1. P. 301–312.
6. Кубенский А.А. Структуры и алгоритмы обработки данных. Объектно-ориентированный подход и реализация на C++. СПб.: БВХ-Петербург, 2004. 464 с.
7. Гасфилд Д. Строки, деревья и последовательности в алгоритмах. Информатика и вычислительная биология. СПб.: Невский Диалект, БВХ-Петербург, 2003. 656 с.
8. Кнут Д.Э. Искусство программирования. Т. 2. Основные алгоритмы. 3-е изд. М.: Вильямс, 2000. 832 с.
9. Кормен Т.Х., Лейзерсон Ч.И., Ривест Р.Л., Штайн К. Алгоритмы: построение и анализ. 3-е изд. М.: Вильямс, 2013. 1328 с.
10. Богомолов А.М., Салий В.Н. Алгебраические основы теории дискретных систем. М.: Физматлит, 1997. 368 с.
11. Кремер Н.Ш. Теория вероятностей и математическая статистика. 2-е изд. М.: Юнити-Дана, 2003. 573 с.
12. Srivastava N., Mishra V., Bhattacharya A. Analyzing the

References

1. Tai K.-C. The tree-to-tree correction problem. *Journal of ACM*, 1979, vol. 26, no. 3, pp. 422–433. doi: 10.1145/322139.322143
2. Zhang K., Shasha D. Simple fast algorithms for the editing distance between trees and related problems. *SIAM Journal of Computing*, 1989, vol. 18, no. 6, pp. 1245–1262.
3. Demaine E.D., Mozes S., Rossman B., Weimann O. An optimal decomposition algorithm for tree edit distance. *ACM Transactions on Algorithms*, 2009, vol. 6, no. 1, art. 2. doi: 10.1145/1644015.1644017
4. Pawlik M., Augsten N. RTED: a robust algorithm for the tree edit distance. *Proc. 38th VLDB Endowment*. Istanbul, Turkey, 2012, vol. 5, no. 4, pp. 334–345.
5. Augsten N., Boehlen M., Gamper J. Approximate matching of hierarchical data using pq-grams. *Proc. 31st Int. Conf. on Very Large Data Bases*. Trondheim, Norway, 2005, vol. 1, pp. 301–312.
6. Kubenskii A.A. *Structures and algorithms of data processing. Object-oriented approach and implementation on C++*. St. Petersburg, BHV Publ., 2004, 464 p. (In Russian)
7. Gusfield D. *Algorithms on String, Trees, and Sequences. Computer Science and Computational Biology*. Cambridge University Press, 1997, 556 p.
8. Knuth D.E. *The Art of Computer Programming. Vol. 2. Seminumerical Algorithms*. Addison-Wesley, 1998.
9. Cormen T.H., Leiserson C.E., Rivest R.L., Stein C. *Introduction to Algorithms*. 3rd ed. MIT Press, 2009, 1312 p.
10. Bogomolov A.M., Salii V.N. *Algebraic Foundations of the Discrete Systems Theory*. Moscow, Fizmatlit Publ., 1997, 368 p. (In Russian)
11. Kremer N.Sh. *Probability Theory and Mathematical Statistics*. 2nd ed. Moscow, Yuniti-Dana, 2003, 573 p. (In Russian)

sensitivity of pq-gram distance with p and q // Proc. 10th Int. Conf. on Very Large Data Bases. Singapore, 2010.
13. Wagner R.A., Fischer M.J. The string-to-string correction problem // Journal of ACM. 1974. V. 21. N 1. P. 168–173. doi: 10.1145/321796.321811
14. Захаров В.Н., Хорошилов А.А. Автоматическая оценка подобия тематического содержания текстов на основе сравнения их формализованных смысловых описаний // Труды 14-й Всероссийской научной конференции «Электронные библиотеки: перспективные методы и технологии, электронные коллекции», RCDL-2012. Переславль-Залесский, Россия, 2012.
15. Хантер Д., Рафтер Д., Фаусетт Д., ван дер Влист Э. и др. XML. Работа с XML. 4-е изд. М.: Диалектика, 2009. 1344 с.
16. Флах П. Машинное обучение. Наука и искусство построения алгоритмов, которые извлекают знания из данных. М.: ДМК Пресс, 2015. 400 с.

12. Srivastava N., Mishra V., Bhattacharya A. Analyzing the sensitivity of pq-gram distance with p and q. *Proc. 10th Int. Conf. on Very Large Data Bases*. Singapore, 2010.
13. Wagner R.A., Fischer M.J. The string-to-string correction problem. *Journal of ACM*, 1974, vol. 21, no. 1, pp. 168–173. doi: 10.1145/321796.321811
14. Zakharov V.N., Khoroshilov A.A. Automatic assessment of similarity of the texts' thematic content on the base of their formalized semantic descriptions comparison. *Proc. RCDL-2012*. Pereslavl'-Zalesskii, Russia, 2012.
15. Hunter D., Rafter J., Fawcett J., van der Vlist E. et. al. *Beginning XML*. 4th ed. Wiley, 2007, 1080 p.
16. Flach P. *Machine Learning: The Art and Science of Algorithms That Make Sense of Data*. Cambridge University Press, 2012, 409 p.

Авторы

Андреева Александра Георгиевна – студент, Университет ИТМО, Санкт-Петербург, 197101, Российская Федерация, Tiari@mail.ru

Маркина Татьяна Анатольевна – кандидат технических наук, доцент, Университет ИТМО, Санкт-Петербург, 197101, Российская Федерация, tmark812@mail.ru

Authors

Alexandra G. Andreeva – student, ITMO University, Saint Petersburg, 197101, Russian Federation, Tiari@mail.ru

Tatiana A. Markina – PhD, Associate Professor, ITMO University, Saint Petersburg, 197101, Russian Federation, tmark812@mail.ru