012-Buzikashvili_H:_Layout 1 12.05.2008 22:21 Page 75

МНОГОЗАДАЧНЫЙ ПОИСК: ФАКТ, АРТЕФАКТ, ПРЕНЕБРЕЖИМОЕ ИСКЛЮЧЕНИЕ?

MULTITASKING SEARCH: FACT, ARTIFACT, OR NEGLIGIBLE EXCEPTION?

Бузикашвили Н.Е. *(buzik@cs.isa.ru)*
Институт системного анализа РАН

Рассмотрено, как часто и в какой форме пользователь поисковой машины Интернета выполняет несколько поисковых задач параллельно. Показано, что параллельное исполнение очень редко и ограничивается двумя задачами, выполняемыми как вложенные. Характеристики временных сессий, содержащих одну, несколько последовательно и несколько параллельно исполняемых задач, различны, а наиболее простым из трех классов оказались последовательно исполняемые задачи.

1. Введение

Практические ситуации, в которых до сих пор изучалась многозадачность – это, прежде всего, ситуации *вынужденной* многозадачности, когда, оператор, в силу внешних обстоятельств, принужден контролировать сразу несколько процессов [6,9]. Типичный пример – работа диспетчера. Принципиальное отличие *невынужденной* многозадачности – принятие самим оператором решения о параллельном исполнении задач. Это решение может быть связано с ситуативной оценкой ожидаемых затрат и рисков, а может носить “ритуальный” характер или иметь форму автоматизма.

Большинство лабораторных исследований показывает, что многозадачность более затратна и менее эффективна, чем последовательное выполнение тех же задач [6,7,9], и невынужденная многозадачность априори должна быть редкой. Поиск в Интернете относится как раз к этой категории. Поэтому, выводы [11–14] о массовом характере многозадачности, согласно которым многозадачными являются 11-12% поисковых сессий пользователей Интернета, достаточно неожиданны. С другой стороны, легко найти (пусть и менее частые) ситуации, когда многозадачность хотя и более затратна, но *приемлема* для исполнителя, либо когда затраты на параллельное исполнение задач *не выше*, чем на последовательное. Поэтому результаты [11–14] могут быть верны. В таком случае, нужно отнести поиск к числу немногих сфер, где многозадачность приемлема.

В [11–14] разметка запросов пользователя на задачные сессии выполнялась вручную, а критерий отнесения разных запросов к одной задачной сессии не указан. Поэтому вероятная ценность [11–14] ограничивается самим фактом обнаружения многозадачности, использованный же метод (ручная разметка) едва ли представляет интерес. Если многозадачность редка, необходимо использовать автоматическую или, как минимум, полуавтоматическую процедуру. Ее суть состоит в том, чтобы на первом, автоматически выполняемом шаге исключить из дальнейшей обработки сессии, почти наверняка не являющиеся многозадачными, после чего асессор выбрал бы среди сессий, автоматически опознанных как многозадачные, те, что действительно являются ими. Сессии, автоматически не опознанные как многозадачные, асессору не предъявляются. Такая процедура эффективна, только если на автоматическом шаге *ошибка отбраковки* (опознание многозадачной сессии как немногозадачной) мала, а доля отбракованных сессий – велика. Последнее необходимо для многократного сокращения затрат асессора. Величина *ошибки неотбраковки* (ложного опознания сессии как многозадачной) не важна, т.к. эти ошибки, в отличие от ошибок отбраковки, устраняются на этапе ручной проверки.

Построенная нами автоматическая процедура отбраковывала как не-многозадачные порядка 99% всех временных сессий. На этапе выборочной ручной проверки только чуть более половины из них были признаны действительно многозадачными.

Были сформулированы гипотезы и вопросы о том, как пользователь выполняет многозадачные сессии (сколько задач может исполняться параллельно, как чередуется их исполнение, как возобновляются прерванные задачи), которые были проверены автоматически, на полных наборах данных. Ручной контроль подтвердил выводы, полученные при автоматической обработке.

Кроме того, рассмотрены числовые характеристики трех классов сессий: (*а*) однозадачные сессии, (*б*) линейные, в которых задачи исполняются последовательно одна за другой, и (*в*) собственно многозадачные, в

012-Buzikashvili_H:_Layout 1 12.05.2008 22:21 Page 76

которых одни задачи временно прерываются для исполнения других. Оказалось, что характеристики этих классов различны. По их значениям можно судить о сложности исполнения каждого из классов сессий. То, что многозадачные сессии наиболее трудоемки, не удивительно. Но также оказалось, что сложность задачи в однозадачной сессии выше, чем у задачи в линейной сессии.

2. Многозадачность: исполнительные цепочки, разделяемые ресурсы

Термин “многозадачный поиск” применительно к поисковому поведению пользователя Интернета был впервые использован А.Спинк [11–14]. Что понимают под многозадачностью в смежных с исследованием поискового поведения областях? Почти полвека назад “многозадачный режимом” была названа такая работа операционной системы, при которой несколько задач выполняются “одновременно”. Операционные системы работали на однопроцессорных машинах, и многозадачный режим был синонимом режима с разделением времени, при котором несколько задач, чередуясь, использовали общий ресурс. Совершенно в этом же смысле многозадачность используется в когнитивной и инженерной психологии. Последовательное выполнение задач не называется многозадачностью и содержательно эквивалентно однозадачности.

Говоря о многозадачности, имеют в виду одну из двух форм: (*а*) исполнение разных задач в один и тот же момент времени или (*б*) чередование задач по схеме прерывание задачи – выполнение другой – ее прерывание и возобновление ранее прерванной.

Компьютерная (=физиологическая) метафора делают вопрос о многозадачности совершенно прозрачным и позволяют обойти неплодотворную психологическую дискуссию “многозадачность – сложнее или проще?”. Есть набор *ресурсов* (процессоры, разделы мозга, периферийные устройства и органы), используемых в задачах. Задаче соответствует *исполняющая цепочка* – последовательность используемых ею ресурсов. Если цепочки не пересекаются, то задачи могут исполняться одновременно, что не ведет к увеличению затрат. Однако если же две цепочки используют один и тот же ресурс, то возникает проблема разделения этого ресурса. Именно ситуации пересечения цепочек типичны для лабораторных исследований многозадачности. Заметим, что и в этом случае поведение оператора скорее описывается не критерием минимизации затрат, а критерием их приемлемости, т.е. не превышают ли они некоторый порог.

Одновременное исполнение непересекающихся задач, когда оно возможно, часто проще и приятнее чередования. Напротив, пересечение (обычно совпадение) исполняющих цепочек может сделать совокупность относительно простых задач невыполнимой, как например, в сеансе одновременной игры вслепую.

В терминах исполняющих цепочек, многозадачный поиск – это тотальное разделение ресурсов: все поисковые задачи используют одни и те же ресурсы, а несовпадение исполняющих цепочек возможно только в конечном звене – окне поиска. Поскольку современная поисковая среда обеспечивает возможность выполнять несколько задач параллельно в разных окнах, препятствием поисковой многозадачности могут быть только ресурсы самого пользователя, но не поисковый интерфейс.

Воспользуемся еще одной, приблизительной метафорой. Когда человек несколько раз подбросит шарик ○, а затем несколько раз второй ●, мы не говорим о жонглировании, а затраты здесь те же, как при подбрасывании одного шарика. Однако такое “псевдожонглирование” есть точный аналог последовательного исполнения нескольких задач. Каков в действительности порядок выполнения транзакций разных задач при их параллельном исполнении? Случайный, регулярный и если регулярный, то что это за регулярность. Сколькими задачами может одновременно “жонглировать” пользователь? На Рис. 1 изображены все варианты “жонглирования” двумя поисковыми задачами, содержащими 2 и 3 транзакции. Определяющее сложность исполнения число возобновлений незавершенной задачи равно 0 для перестановок, соответствующих последовательному исполнению. Следующее по сложности – вложенное исполнение, в котором одна задача, прервав другую, исполняется до конца, и самое сложное – смена задачи после каждой транзакции (“истинное жонглирование”).

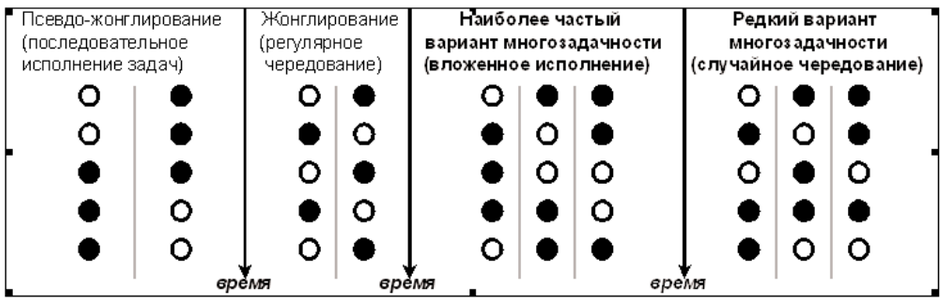


Рис. 1. “Жонглирование” двумя задачами

012-Buzikashvili_H:_Layout 1 12.05.2008 22:21 Page 77

3. Определения

В статье используются следующие понятия:

(1) *транзакция* – акт взаимодействия пользователя с поисковой машиной. Транзакция может соответствовать либо *подаче запроса*, либо *листанию* страницы (экрана) результатов запроса. Для описания транзакции будем использовать пару (*запрос, страница_результатов*). Например, если пользователь (1) подал запрос и получил нулевую страницу результатов (1-я транзакция), (2) перелистнул результаты на следующую страницу (2-я транзакция), (3) вернулся к нулевой странице (3-я транзакция), то последовательность описаний транзакций: (*запрос*, 0), (*запрос*, 1), (*запрос*, 0).

(2) *временная сессия* – последовательность транзакций одного пользователя с поисковой машиной, отделенная от предыдущих и последующих его транзакций временем, не меньшим используемым в исследовании *межсессионным временным порогом*. В разных исследованиях этот порог варьируется от 5 мин [10] до 2 часов [8]. В работах [11–14] использовалось 15-минутное значение.

(3) *задача* – поиск одного и того же объекта или очень сходных объектов;

(4) *задачная сессия* – все транзакции временной сессии, относящиеся к одной задаче. Ниже, для краткости и во избежание путаницы с временной сессией, будем называть задачей именно задачную сессию (что путаницы не повлечет).

(5) рассматриваемые классы временных сессий:

(5a) *однозадачная сессия* (все запросы временной сессии относятся к одной задаче),

(5b) *линейная сессия* из нескольких задач содержит более одной задачи, которые выполняются одна за другой без прерываний незавершенных задач,

(5c) *многозадачная сессия* содержит несколько задач, среди которых хотя бы одна прерывается другой.

(6) Текущая ширина временной сессии – число незавершенных (первая транзакция уже выполнена, а последняя – еще нет) на данный момент (на момент очередной транзакции) задач. *Шириной сессии* назовем ее максимальную текущую ширину.

4. Вопросы и гипотезы

В исследовании мы рассмотрели следующие вопросы и гипотезы (см. также [2,3]):

(*a*) Сколь часто различные задачи выполняются параллельно, какова доля временных сессий, содержащих участки параллельного выполнения двух и более задач?

(*b*) Каково соотношение двух типов возобновления прерванной задачи, т.е. как часто задача возобновляется: 1) вводом нового запроса (точнее, транзакции с 0-й страницей результатов) или 2) листанием результатов запроса, введенного до прерывания?

Гипотеза о возобновлении состоит в том, что поскольку продолжение работы с результатами запроса, поданного перед прерыванием, более затратно, чем подача нового запроса, доля возобновлений-листаний меньше доли листаний среди продолжений линейной сессии.

(*c*) *Гипотеза о схемах переключения* между незавершенными задачами. Если имеется несколько незавершенных задач, то за транзакцией одной задачи, скорее всего, последует ее же транзакция. В частности, это значит, что порядок исполнения “остающихся на данный момент” транзакций незавершенных задач отличен от случайной перестановки.

(*d*) Гипотеза о числовых характеристиках поискового поведения: характеристики параллельно исполняемых задач (длина сессии, длина запроса, число транзакций на запрос) отличаются от характеристик задач, исполняемых линейно одна за другой. Значения числовых характеристик можно интерпретировать в терминах сложности задач и сессий. Предположительно, использование нескольких задач в одной временной сессии более вероятно, если задачи простые. При этом параллельное исполнение сложнее линейного исполнения тех же задач.

(*e*) Естественную гипотезу, что запросы параллельных задач сделаны из разных окон, к сожалению, проверить нельзя, т.к. логи не содержат полей *окно* (*экземпляр браузера*).

5. Наборы данных

Мы использовали логи транзакций двух поисковых машин: (1) почти полностью русскоязычный лог *Яндекса* (недельная выборка, 175000 пользователей, 2005); (2) два набора из почти полностью англоязычного лога *Excite* – полный 8-часовой набор, 1999, 537600 пользователей и суточная выборка 2001, 305000 пользователей. Логи *Excite* многократно использовались ранее, в том числе в исследованиях многозадачности [14], и взяты, чтобы, с одной стороны, сравнивать результаты, полученные на одних и тех же данных в разных исследованиях,

012-Buzikashvili_H:_Layout 1 12.05.2008 22:21 Page 78

а с другой – сравнить результаты, полученные на разных данных одним и тем же инструментом.

Автоматическая предобработка каждого из наборов данных состояла в:

— исключении клиентов, предположительно являющихся роботами. С помощью скользящего временного окна [4] исключались клиенты, сделавшие более 7 *разных* запросов в течение одного часа.

— сегментации последовательности транзакций, выполненных одним пользователем (опознанным как робот), на временные сессии. Для сопоставимости с результатами [11–14], использовался тот же межсессионный порог (15 мин), что и в [11–14].

6. Метод

Суть метода, который был применен для выделения задач внутри временных сессий одного пользователя, проста:

(0) запросы, присутствующие в транзакциях одной временной сессии, учитываются однократно. Напр., в сессии (*кот*, 0) (*кот*, 1) (*пес*, 0) есть только 2 запроса – *кот* и *пес*.

(1) для каждой пары запросов, относящихся к одной временной сессии, определяется их сходство: если два запроса содержат хотя бы одно общее слово, являющееся основной частью речи (основными считались существительные, глаголы, прилагательные, причастия и флективно совместимые с ними неизвестные слова), то запросы сходны. Например, <*этот кот*> и <*серый кот* > сходны, а <*этот дом*> и <*этот кот*> – нет.

Однако в запросах, с одной стороны, массовы ошибки (пропуски и перестановки букв, набор одной вместо другой, склейки и вставки пробелов), причем значительная часть слов запросов – не словарные. С другой стороны, какие-то слова просто не имеют однозначного написания, например, иноязычные имена и названия. Чтобы покрыть ошибки и неоднозначности, шаг (1) определения сходства запросов был заменен на (1*), основанный на общих подстроках [1]. Похожий и тем же мотивированный метод использован в [5].

(1*) для всех запросов, относящихся к одной временной сессии строятся парные исходному запросу строки: символы исходного запроса переводятся в строчные, пробелы и пунктуации отбрасываются. Так, запросу <*Жозеф-Луи +Гей-Люссак*> будет сопоставлена строка *жозефлуигейлюссак*. Для каждой пары таких строк определяется бинарное орфографичекое сходство. Именно, если две строки содержат одинаковые подстроки не короче l=3 символов, то мы пытаемся посимвольно нарастить их в каждой строке слева и справа с тем, чтобы выделить сходные подстроки. При наращивании допускается 4 типа замен: перестановка соседних символов, вставка и удаление одного символа, замена символа. Применив исправление к одной из текущих сходных подстрок, мы не имеем права применять его к следующим thr=3 символам с этого же края подстроки. Когда наращивание схожих подстрок завершено, смотрим, покрывает ли в исходных запросах хотя бы одна из подстрок хотя бы нефлективную часть хотя бы одного слова, являющегося основной частью речи. Если да, запросы считаются сходными. Напр., запросы <*царствующий дом*> и <*шествующий кот*> несходны, т.к. их общая часть *ствующий* приходится на флексию, тогда как запросы <*уфимский богатырь*> и <*сы руфмскйи*> опознаются как сходные. Критерий

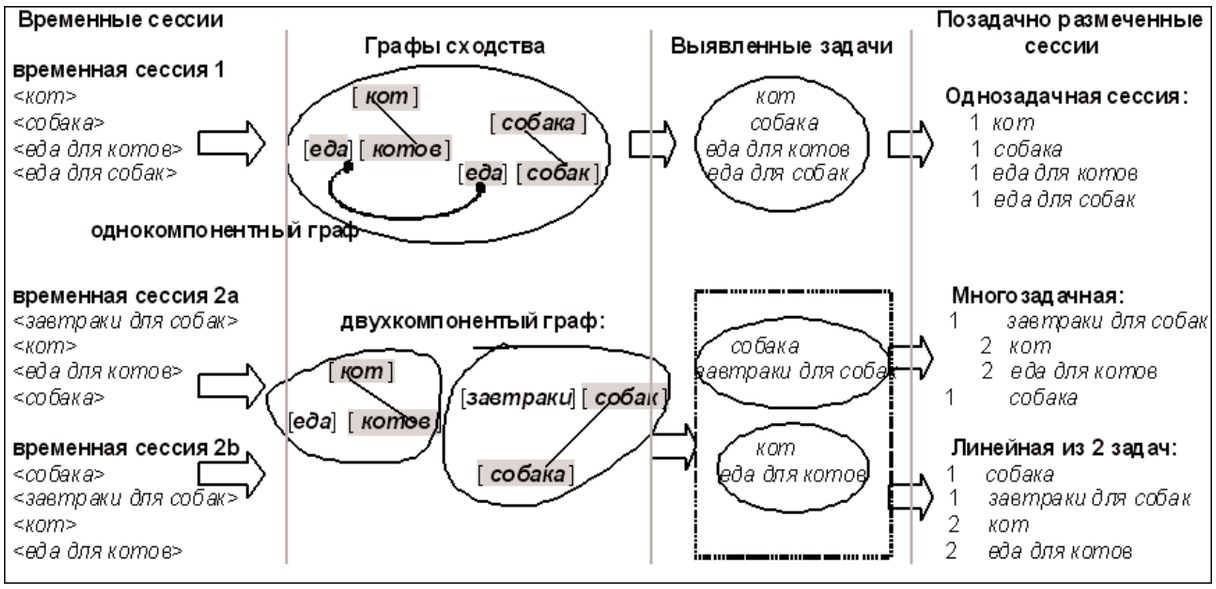


Рис. 2. Примеры выявления задач в трех временных сессиях

012-Buzikashvili_H:_Layout 1 12.05.2008 22:21 Page 79

сходства устойчив к инверсиям, пропускам и вставкам букв, к лишним пробелам, пунктуациям, а также склейкам. Результат шага (1*) – бинарная матрица парного сходства запросов.

(2) После построения матрицы сходства запросов одной временной сессии строится транзитивное замыкание этой матрицы, и запросы, попавшие в одну компоненту связности, считаются относящимися к одной задаче.

(3) Наконец, определяется ширина временной сессии и если сессия широкая, она отписывается в файл сессий данной ширины для последующей ручной проверки.

На Рис. 2 приведен пример работы всей процедуры, а на Рис. 3 – фрагмент результата работы метода выявления задачных сессий (числа слева – маркировка транзакций номерами задач, а положение номеров соответствует уровню вложенности задачи). Как видим, во втором примере (user 119, time session 2) на Рис. 3 сходство ***хлеб****опечения* и *архангельск****хлеба*** не обнаружено, в результате чего ошибочно выявлена задачная сессия 4, в действительности относящаяся к сессии 2.

```
user 108, t.sess. 9
1       программа анонимности в интернете
   2    ростовская доска объявлений
1       лохотрон
1       интернет лохотрон
--------
user 119, t.sess. 2
1       стихи р. рождественского
1       стихи роберта рождественского
2       журнал хлебопечение россии
   3    рамблер
2       хлебопечение россии подколзин        сходство
2       союз пекарей подколзин               не обнаружено
2       подкозин фотографии
   4    архангельскхлеб
   4    буклет архангельскхлеба
2      фотогрфия сергея николаевича подколзина
```

Рис. 3. Фрагмент файла временных сессий ширины 2

Описанная процедура не позволяет выявить сходство орфографически различных, но содержащих синонимы запросов. Однако гораздо чаще классической словарной синонимии в запросах встречается “неклассическая”: (1) “субкультурная синонимия”, т.е. сленг, часто узкогрупповой; (2) “многоязычие”: одно и то же слово дано в запросах то по-русски, то по-английски; (3) ошибки переключения языка; (4) использование наравне с полными формами аббревиатур, а самое интересное – (5) задание не сходных, а *совместимых* признаков, характеризующих разные свойства искомого.

Далее, сходство запросов процедура выявляет систематически реже, а совместимость не выявляет вообще. Это завышает число задач и влечет ложное обнаружение многозадачности. Однако можно воспользоваться систематическим завышением многозадачности как преимуществом.

Именно, после автоматического выявления предположительно многозадачных сессий нужно вручную проверить лишь эти сессии, доля которых составляет порядка 1% (Табл. 1). Т.е. асессор вместо полной разметки 100 сессий должен обработать одну. При этом, при обработке сессии, автоматически уже размеченной на задачи, асессор должен лишь выяснить сходство задач, а не заново разметить все транзакции как в [11–14].

7. Результаты автоматического анализа многозадачного поиска

Некоторые результаты автоматической обработки приведены в Табл. 1. Заметим, что хотя наборы данных относятся к разным языкам, разным поисковым машинам и разным годам, одноименные результаты на всех наборах сходны, что, как надеемся, отражает не влияние метода, а сходство поискового поведения пользователей.

В любой ситуации пользователи выбирают наименее затратную тактику:

(*a*) избегают жонглироть задачами (лишь ~1% временных сессий многозадачные);

(*b*) когда пользователь все же “жонглирует” задачами, он ограничивается 2 “шарами” (в т.ч. когда временная сессия содержит более двух задач), и при этом чаще всего избирает

(*c*) наименее затратный, вложенный вариант многозадачности, при котором число возвратов к прерванной задаче минимально. Именно, прервав одну задачу, он целиком, от начала и до конца выполняет вторую, и лишь

012-Buzikashvili_H:_Layout 1 12.05.2008 22:21 Page 80

завершив ее, возвращается к прерванной. Гипотеза о форме возобновлении не подтвердилась – доля новых запросов как продолжений прерванной задачи та же, что при линейном исполнении, т.е. пользователь часто возвращается к листанию результатов запроса, сделанного перед прерыванием.

(*d*) Число запросов в широких временных сессиях больше, чем в сессиях, в которых задачи исполняются последовательно. Более того, и число листаний результатов одного запроса (т.е. число транзакций на запрос) для широких сессий больше. В то же время запросы широких временных сессий короче.

Классы временных сессий:	**Однозадачные сессии**			**Линейные сессии (задачи исполняются одна за другой)**			**Многозадачные (задачи исполняются параллельно)**		
Набор данных	*Excite* 1999	*Excite* 2001	*Яндекс* 2005	*Excite* 1999	*Excite* 2001	*Яндекс* 2005	*Excite* 1999	*Excite* 2001	*Яндекс* 2005
% временных сессий, отнесенных к классу	87.56	86.87	86.40	11.32	11.94	12.53	1.12	1.19	1.07
Число задач во вр. сессии	1	1	1	2.20	2.21	2.19	2.34	2.35	2.30
Возобновление задачи новым запросом (%)	27.34	26.8	26.9	54.60	52.4	53.1	51.84	52.8	52.4
Число (несовпадающих) запросов в задаче	1.24	1.3	1.3	1.17	1.2	1.2	1.45	1.5	1.5
Число (несовпадающих) запросов во врем. сессии	1.24	1.3	1.3	2.56	2.6	2.6	3.40	3.5	3.4
Число транзакций одного запроса	1.56	1.7	1.6	1.52	1.6	1.5	1.87	2.0	2.0
Число транзакций в задаче	1.93	2.13	2.09	1.78	1.85	1.83	2.71	2.97	2.92
Продолжительность (мин) временной сессии:									
из 1 запроса	1.09	1.17	1.68						
из 2 запросов	4.37	4.45	5.60	5.84	5.72	6.89	10.62	11.52	13.55
из 3 запросов	7.35	7.36	8.97	9.27	8.87	11.04	11.16	11.79	14.30

Таблица 1. Результаты автоматического анализа трех логов

(*d'*) Качественное поведение характеристик в зависимости от типа временной сессии приведено на Рис. 4. На первый взгляд, наименее сложными в исполнении должны быть однозадачные сессии, сложность линейных сессий должна быть не ниже, и, наконец, параллельное исполнение задач наиболее сложно. Однако немонотонное поведение числовых характеристик на Рис. 4, говорит в пользу того, что субъективная сложность задач в однозадачных сессиях выше, чем у задач в линейных сессиях. Т.е. если текущая задача сложна для пользователя, то, скорее всего, он не начнет следующую.

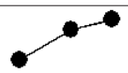
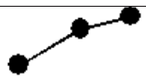
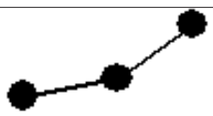
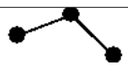
Характеристики, “монотонно” изменяющиеся по 3 классам временных сессий:		Характеристики, “немонотонно” изменяющиеся по 3 классам временных сессий:	
Число задач во временной сессии		Число несовпадающих запросов в задаче	
Число (уникальных) запросов в сессии		Число транзакций запроса	
Продолжительность сессии		Термов в запросе	
		Возобновление задачи новым запросом	

Рис. 4. Диаграммы изменения количественных характеристик по 3 классам временных сессий: левая из 3 точек соответствует значению характеристики на однозадачных сессиях, центральная – сессиям последовательного исполнения, правая – сессиям параллельного исполнения

012-Buzikashvili_H:_Layout 1 12.05.2008 22:21 Page 81

8. Ручная проверка

Была проведена выборочная проверка около 300 временных сессий, автоматически опознанных как многозадачные. Оказалось, что

(1) почти половина из них многозадачными не являются. Лексическое несходство запросов вызвано активным поиском, в котором пользователь, выполняя одну задачу, радикально меняет запросы и возвращается к запросам, похожим на ранее сделанные.

(1.1) среди временных сессий, автоматически опознанных как сессии ширины 2, настоящих многозадачных сессий чуть больше половины, и все они ширины 2;

(1.2) зато среди временных сессий, опознанных как более широкие, оказалось лишь несколько ширины 2, а все остальные были сессиями активного однозадачного поиска.

(2) характеристики действительно многозадачных сессий очень близки к характеристикам сессий, автоматически опознанных как многозадачные;

(3) среди действительно многозадачных сессий прерывающая задача допускает интерпретацию как связанная с прерванной (как ее ответвление или как поиск того, без чего продолжение прерванной затруднительно) редко (менее 20%). Обычно же прерванная и прерывающая задачи никак не связаны.

9. Заключение

Показано, что:

— многозадачность, понимаемая параллельное исполнение нескольких поисковых задач, – крайне редкая манера поиска;

— ширина многозадачных сессий равна двум, т.е. в любой момент есть не более одной начатой, но не завершенной поисковой задачи;

— возврату к прерванной задаче предшествует завершению ее прервавшей;

Таким образом, поисковое поведение пользователя подчиняется принципу минимума усилий: пользователь избегает многозадачности как более затратной, а прибегнув к ней, ограничивается исполнением только двух задач, причем наиболее экономным образом.

Параллельный многозадачный поиск, безусловно, факт, но столь редкий, что нет никакой необходимости в дальнейшем его изучении, по крайней мере, в контексте работ [11–14, 2,3]. Цель данной работы локальна – показать ошибочность мнения о массовости многозадачного поиска. Метод, использованный здесь, предназначен для решения именно этой задачи, которое никак не переносимо за ее рамки. Очень высокое число ошибок неотбраковки, не вызывающее затруднений в использованной процедуре, делает невозможным применение метода для автоматического решения другой, гораздо более важной проблемы – вычленения задач в последовательности всех запросов пользователя.

10. Благодарности

В заключение автор считает долгом выразить признательность Ю.Г. Зеленкову, Н.В. Пономаревой, А.В. Сокирко и А. Спинк за полезные обсуждения, а также анонимному рецензенту (в том числе, если он уже упомянут) – за ценные замечания по проблематике выявления задачных сессий.

Список литературы

1. Apostolico A.: String editing and longest common subsequences // Handbook of Formal Languages, vol, 2, Springer-Verlag, 1997, 361-398

2. Buzikashvili N. Automatic task detection in the Web logs and analysis of multitasking // 9th Int. Conf. on Asian Digital Libraries (ICADL 2006), LNCS 4312, Springer Verlag, 2006, 131-140.

3. Buzikashvili N., Ponomareva N.V. Multitasking search on the Web // 7th Int. Conf. on Cognitive Modeling (ICCM 2006). Edizioni Goliardiche, 2006, 357-358.

4. Buzikashvili N. Original and normalized Web log metrics as functions of controllable variables of log study // 5th Latin American Web Conference (LA-WEB 2007), IEEE CS, 2007, 3-12.

5. Kaki M. fKWIC: Frequency-based keyword-in-context index for filtering Web search results // JASIST, 52(12), 2006, 1606-1615

6. Kushleyeva Y., Salvuci D, Lee FJ. Deciding when to switch tasks in time-critical multitasking // Cognitive Systems Research, 6, 2005, 41-49.

012-Buzikashvili_H:_Layout 1 12.05.2008 22:21 Page 82

7. Monsell S.: Task switching // Trends in Cognitive Sciences, 7(3), 2003, 134-140

8. Montgomery A., Faloutsos C. Identifying Web browsing trends and patterns // Computer 34 (7), 2001, 94-95.

9. Rubinstein, J., Meyer, D., Evans, J. Executive control of cognitive processes in task switching // J. Exp. Psycol. Hum. Percept. Performance, 27, 2001, 763-797.

10. Silverstein, C., Henzinger, M., Marais, H., Moricz, M.: Analysis of a very large web search engine query log // SIGIR Forum, 33 (1), 1999,: 6–12.

11. Spink A., Jansen B.J., Pedersen J. Multitasking Web search on Alta Vista // Int. Conf. on Information Technology: Coding and Computing (ITCC'04). IEEE CP. 2004, 309-313.

12. Spink A., Jansen B.J., Park M., Pedersen J. Multitasking during Web search sessions // Information Processing & Management, 42(1), 2006, 264-275.

13. Spink A, Koshman S, Park M, Field C, Jansen B.J. Multitasking Web Search on Vivisimo.com // Int. Conf. on Information Technology: Coding and Computing (ITCC'05), Vol. II, IEEE CS, 2005, 486-490.

14. Spink A., Cole C., Waller M. Multitasking behavior. // Chapter 4. Annual Review of Information Science and Technology (ARIST) vol. 42, 2008, Medford, NJ: Information Today, Inc. (to appear).