Литература

1. Amazon Web Services home page. http://aws.amazon.com/.
2. Программная платформа eucalyptushttp://www.eucalyptus.com/.
3. Eucalyptus Public Cloud (EPC) http://eucalyptus.cs.ucsb.edu/wiki/EucalyptusPublicCloud/.
4. Фингар П. Dot.Cloud: облачные вычисления - бизнес-платформа XXI века: Аквамариновая книга. Б.м., 2011. 253 с.
5. Риз Дж. Облачные вычисления: СПб., БХВ-Петербург, 2011 288 с.

Статья поступила 09 12 2011

К.т.н., доц. В.М. Михелев, К.В. Кузнецов, Д.А. Торопчин
(НИУ «БелГУ»)

V.M. Mikhelev, K.V. Kyznetsov, D.A. Toropschin

ПАРАЛЛЕЛЬНЫЕ ВЫЧИСЛЕНИЯ ПРИ РАСПОЗНАВАНИИ НЕЧЕТКИХ ТЕКСТОВЫХ ДУБЛИКАТОВ НА ОСНОВЕ ТЕХНОЛОГИИ CUDA

PARALLEL COMPUTING OF FUZZY RECOGNITION OF DUPLICATE TEXT BASED OF TECHNOLOGY CUDA

Проведено исследование нескольких основных методов определения текстовых дубликатов, основанных на построении частотных словарей документов или коллекций и алгоритмов семейства шинглов Предложен новый алгоритм распознания нечетких текстовых документов, позволяющий использовать параллельные вычисления на основе технологии CUDA

A study of several basic methods for measuring text duplicates based on building of frequency dictionary documents or collections and algorithms family of shingles A new algorithm of fuzzy recognition text documents allows you to use parallel computing based on CUDA technology

Ключевые слова параллельное программирование, текстовый дубликат метод шинглов, технология CUDA

Key words parallel computing, duplicate text, computer recognition method of shingles, cuda technology

Введение

Рост объемов информации в последние годы обуславливает

развитие различных информационных методов поиска, удовлетворяющих заранее определенному условию. Одной из таких задач является задача определения схожести различных текстовых документов между собой.

Существующие методы поиска нечетких дубликатов позволяют с различной степенью уверенности распознать схожие документы, однако их эффективность, в условиях насыщенных различной информацией страниц современного Интернета, позволяет желать лучшего. Кроме того, большинство реально используемых алгоритмов решения этой задачи требует больших временных затрат, поэтому актуальна задача построения алгоритмов поиска нечетких текстовых дубликатов с использованием параллельных вычислений в вычислительных распределенных системах высокой производительности.

Под понятием дубликат документа будем понимать точную копию документа, а под понятием «нечеткий дубликат» документа – документ, частично измененный в содержательной части. При этом будем рассматривать понятие «дубликат документа» только с точки зрения поисковой системы, а не пользователя. Поэтому не будем рассматривать такое явление как «копирайтинг», т.е. переписывание текста специально для поисковых систем с использованием других слов, но с сохранением общего смысла. Такой текст для поисковой машины всегда будет оригинальным, т.к. смысл текста компьютеры различать пока не могут.

Теоретические основы

Нами проведено исследование по использованию нескольких основных методов определения дубликатов. Существующие алгоритмы по распознаванию нечетких текстовых дубликатов можно разделить на 2 группы:

1. алгоритмы, учитывающие глобальные свойства коллекции;
2. алгоритмы, учитывающие только локальные свойства документов.

К первой группе относятся алгоритмы, основанные на построении частотных словарей документов или коллекций. Ко второй группе относиться алгоритмы семейства шинглов , Long Sent [1].

На практике довольно часто используется алгоритм шинглов. При использовании этого алгоритма весь исследуемый документ предварительно проходит канонизацию, т.е. приведение текста к удобному для анализа виду. Канонизация текста довольно затратная часть алгоритма, т.к. текст надо отчистить от так называемых стоп слов и стоп символов. В различных реализациях этого алгоритма [2] можно встретить также удаление из текста прилагательных и даже замену слова на самый часто встречающийся синоним. Затем текст нарезается на подстроки определённой длины, называемые шинглами. На следующем этапе выполняется отсеивание повторяющихся шинглов и для каждого шингла из этого множества считается хеш-функция. В различных модификациях алгоритма предлагается брать определённый набор шинглов или объединять их в определённые подмножества. Из наиболее известных модификаций часто используются метод мегашинглов [3].

Главным достоинством алгоритма шинглов является простота его реализации. Однако этот метод не устойчив к небольшим изменениям текста. Например, если изменить одно слово, то приходится изменять n шинглов, где n-длинна щингла.

Алгоритм Long Sent (длинных предложений) состоит в том, что из всего текстового документа выбирается несколько, чаще всего - два, самых длинных предложения и они сцепляются вместе в строку. Затем определяется сигнатура этого документа, как значение хеш-функции от этой строки. Достоинством этого алгоритма является простота его реализации и высокая скорость вычислений Однако этот метод не устойчив к небольшим изменениям текста.

Большая группа алгоритмов основана на использовании частотных словарей. В этой группе можно выделить метод TF*IDF и метод TF*RIDF. Метод TF*IDF основан на том, что выбираются слова с наибольшим весом, который вычисляется по следующей формуле

$$wt = TF * IDF$$

$$IDF = \log(\frac{N - df + 0.5}{df + 0.5})$$

$$TF = \frac{tf}{2 * (0.25 + 0.75(\frac{dl}{dl\_avg})) + tf}$$

где: tf – частота слова в документе,

df – число документов содержащих слово,

dl- длина документа,

dl_avg – средняя длина документа,

N – количество документов.

Метод TF*RIDF основан на том, что слова распределяются в коллекции случайным и независимым образом. При этом веса слов вычисляются по следующей формуле

$$wt = TF * RIDF$$

$$TF = 0.5 + 0.5\frac{tf}{tf\_\max}$$

$$RIDF = -\log(\frac{df}{N}) + \log(1 - \exp(\frac{-cf}{N}))$$

Далее выбираются самые тяжелые слова и они сцепляются в строку, что и является сигнатурой документа.

Так как все рассмотренные алгоритмы требовательны к вычислительным ресурсам, то был предложен алгоритм, который можно легко распараллелить. Этот метод учитывает только локальные свойства документа и его можно разбить на некоторые логические этапы.

Первым этапом является канонизация текста - слова приводятся к начальной форме слова, для существительных и прилагательных - это единственное число именительного падежа, для глаголов - инфинитив. Также из текста удаляются стоп-слова (предлоги, местоимения), в этой реализации алгоритма удалялись и прилагательные.

Далее текст разбивается на части и для каждой части текста рассчитывается следующая матрица A, где вектор A[i]={a1,a2,..,aj,.,an}, aj = номер(порядковый) слова в тексте, i - номер слова в словаре. Затем рассчитывается вес слова

$$wt = \sum_{i=1}^{n-1} \frac{1}{\log(a_{i+1} - a_i)}$$

С помощью этой формулы можно оценить, как слово распределено по тексту. Если слово встречается один раз, то wt=0. Если же слово раскидано по всему тексту, то wt будет меньше, чем если бы слово встречалось в небольшой окрестности.

На следующим этапе отбрасываем слова, у которых wt=0 и

рассчитываем границы, по которым будем прореживать словарь.

$$hb = \frac{len(dictionary) * \max(dictionary)}{\sum wt}$$

$$lb = \frac{\sum wt}{len(dictionary) * \log(1 / \min(dictionaty))}$$

Затем выбираем слова, которые удовлетворяют следующему условию

$$lb \le wt \le hb$$

Из оставшихся слов выбираются пять с наибольшей длиной, это - сигнатура словаря. Словари пересекаются между собой и результат просчитывается как сумма максимальных результатов пересечения. Сигнатурой документа является набор из сигнатур словарей. Этот подход также позволяет проверять документ на включения.

Вычислительный эксперимент

Тестирование проходило на видеокарте G102M, которая имеет восемь мультипроцессоров и полосу пропускания памяти 6.4 ГБ/сек.

Вычислительный эксперимент №1. Сравнение стандартного алгоритма шинглов и предложенного проводились на текстах двух переводов сказки «Алиса в стране чудес» («Алиса в стране чудес», «Аня в стране чудес»). В таблице (табл. 1) приведено сравнение разработанного метода по сравнению с методом шинглов (значения в таблице в процентах) при операциях изменения текста.

Таблица 1

Результат распознавания нечетких текстовых дубликатов

	Метод шинглов	Предложенный метод
Исходный текст	8.17	35 5
Перемещение блоков (очень больших блоков по сравнению с длиной текста)	7.9	34.4
Замена слов	2.2	34 4
Добавление блоков	7.5	34 4

Вычислительный эксперимент №2 Сравнение стандартного алгоритма шинглов и предложенного проводилось на трех коллекциях документов. Коллекции документов включают в себя 75, 40 и 30 пояснительных записок к курсовым работам студентов по дисциплине "Web программирование". На рис.1 приведен средний процент схожести для документа, когда каждый документ сравнивался с каждым (если n документов общее число сравнений составляет $(n-1)^2$) для разных размеров коллекций.

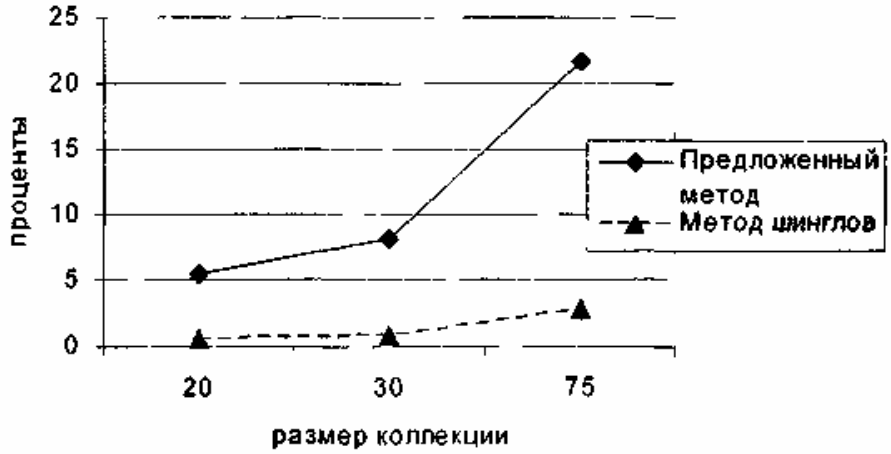

Рис 1

Зависимость среднего процента схожести документов от размера коллекции

На рис.2 приведен процент средний схожести документа с пятью наиболее схожими документами для разных размеров коллекций.

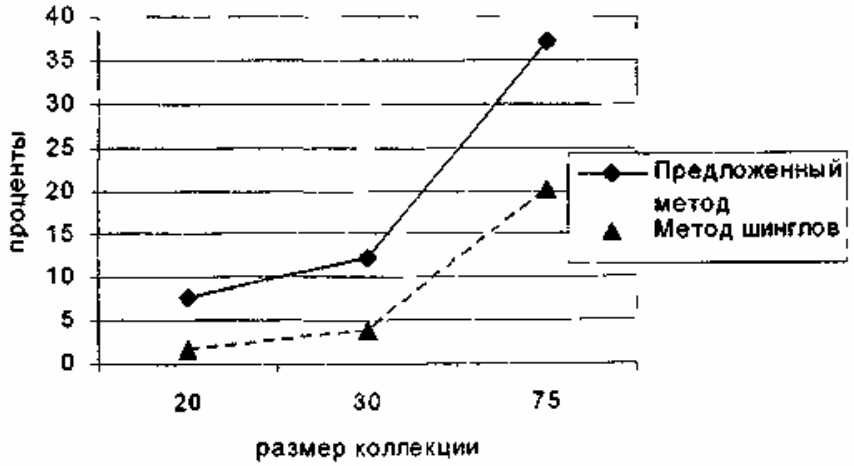

Рис 2

Зависимость среднего процента пяти самых схожих документов в коллекции

На рис.3 приведено время работы алгоритмов для разных размеров коллекций. Можно сделать вывод, что время работы алгоритма шинглов значительно возрастает по сравнению с предлагаемым алгоритмом при увеличении размера коллекций.

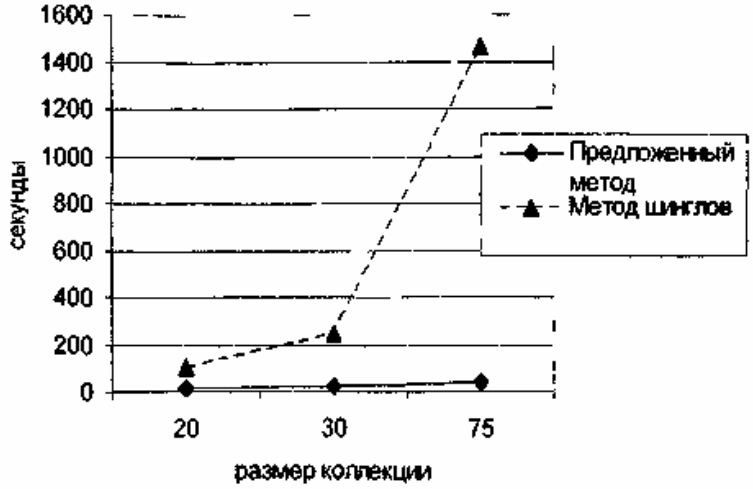

Рис 3
Времени работы алгоритмов при разных размерах коллекций

Выводы

Проведенные исследования показали, что разработанный алгоритм дает лучшие результаты по сравнению с методом шинглов. Параллельные вычисления при распознавании нечетких текстовых документов с использованием технологии CUDA позволили значительно сократить время расчетов

Литература

1 Broder. On the resemblance and containment of documents. Compression and Complexity of Sequences (SEQUENCES'97), - "IEEE Computer Society", 1998, p. 21-29.

2. Неелова Н.В., Сычугов А А Сравнение результатов детектирования дублей методом шинглов и методом Джаккарда. – "Вестник РГРТУ", 2010, № 4, вып. 34, с. 72-78.

3 Зеленков Ю.Г., Сегалович И.В Сравнительный анализ методов определения нечетких дубликатов для WEB-документов – В сб.: Электронные библиотеки: перспективные методы и технологии, электронные коллекции: Труды 9-й Всерос. науч. конф.RCDL´2007. Переславль-Залесский, 2007, т. 1, с. 166-174.

4. Михелев В.М , Кузнецов К В., Батищев Д С., Торопчин Д.А Алгоритм распознавания нечетких текстовых дубликатов с использованием технологии CUDA. – В сб . Компьютерные науки и технологии. Труды Второй Междунар. НТК. Белгород, 2011, с 112-116.

Статья поступила 09 12 2011

Д.т.н., проф. Е.Г. Жиляков, А.В. Болдышев, к.т.н. Е.И. Прохоренко (НИУ «БелГУ»)

E.G. Zhilyakov, A.V. Boldyshev, E.I. Prokhorenko

АЛГОРИТМ СЖАТИЯ РЕЧЕВЫХ ДАННЫХ НА ОСНОВЕ ДВУМЕРНОЙ ОБРАБОТКИ ДАННЫХ

SPEECH DATA COMPRESSION ALGORITHM BASED ON TWO-DIMENSIONAL DATA PROCESSING

В статье изложен новый подход к обработке речевых сигналов, а именно алгоритм двумерной их обработки Такой подход позволяет сократить затраты ресурсов вычислительных систем на разбиение сигнала на фрагменты и пофрагментную обработку Приведены результаты вычислительных экспериментов по оценке степени сжатия исходного речевого сообщения, а также коэффициенты корреляции исходного и восстановленного сообщения

In the article describes a new approach to the processing of speech signals namely the algorithm of two-dimensional processing of speech signals This approach reduces the overhead of computing systems at signal decomposition into fragments and fragmental processing The results of computational experiments to assess the degree of compression of the original voice message, as well as the correlation coefficients of the initial and recovered messages

Ключевые слова сжатие речевых сигналов, двумерная обработка сигналов, информационные системы, телекоммуникационные системы

Key words compression of speech signals, two-dimensional data processing, information systems, telecommunication systems

На сегодня проблема ограниченности ресурсов информационно-телекоммуникационных систем для передачи и хранения речевых данных (пропускная способность канала связи, объем памяти жестких носителей), приводит к необходимости поиска путей их