New problems in stringology

AXA Workshop: Advanced Algorithms on Strings
Istituto Veneto di Scienze, Lettere ed Arti
Venice
June 22, 2016

Andrzej Ehrenfeucht
University of Colorado, Boulder
andrzej@cs.colorado.edu

Motto: ***Nihil novi sub sole***

I’ll talk about two methods that were discussed in the nineteen sixties and seventies but were abandoned as unfeasible:

sorting “unordered” sets, and

finding “interesting words” in an arbitrary text.

Sorting

Let S be a set and $f(x, y)$ be a real valued anti-symmetric function on S, $f(x, y) = -f(y, x)$.

Finding a permutation $P = (p_1, \ldots p_n)$ of S which maximizes the sum $v(P) = \sum\{f(p_i, p_j): i < j\}$, is a nasty optimization problem, so replace it by an n-person game.

Game

Define the value of $P = (p_1, \ldots p_n)$ for a player p_i as $v(p_i) = .5*\sum\{f(p_i, p_j): i \neq j\}$, so $v(P) = \sum v(p_i)$.

A player p_i can move to another position in the permutation only when the move increases the value $v(p_i)$. The game ends when no player can move.

Define the minimal target value for a player,
$t(p_i) = .5\max(\sum\{f(p_i, x): x \neq p_i\}, \sum\{f(x, p_i): x \neq p_i\})$

A permutation P is weakly sorted when, for every p_i, $v(p_i) \geq t(p_i)$.

Theorem 1 (very easy)
There exist weakly sorted permutations that can be reached by the *basic strategy* of each player.

Theorem 2 (easy)
If the values of f are *integers*, and max |f| = m, and the game is played by the *basic strategy*, then the time complexity of playing the game is $O(n^3m)$.

An application

Since anti-symmetric functions are closed under addition, therefore many attributes, f_1, f_2, …, f_k, on S can be weakly sorted together using
$f(x, y) = \sum\{f_i(x, y)\text{: } i = 1, …, k\}$.

And this fact allows us to create a binary tree of attributes, where the attributes in one node are "positively correlated", and the "correlations" between attributes in different nodes are mostly negative.

Generic example

Let $P = (P_1, P_2, P_3, P_4, P_5)$ be weakly sorted. So $v(P) \geq \sum\{t(x), x \text{ in } S\}$.
But some of the values $v(P_1)$, …, $v(P_5)$ can be negative (for example, $v(P_3)$ and $v(P_5)$ are negative).

In such case we partition the set of attributes into $\{f_1, f_2, f_4\}$ and $\{f_3, f_5\}$, and create two new weakly sorted permutations, $Q_1 = (P_1', P_2', P_4')$ and $Q_2 = (P_3',P_5')$.

Problems

Interesting words

We look at *occurrences of words* and *scattered-words* in a *text* written in *alphabet* A.

An example

2081183375909708356697052772947203 2   a text in A = {0, 1, 2, 3, 4, 5, 6, 7, 8, 9}

20*811833759*09708356697052*7729472*032   occurrences of *words*
811833759 and 7729472

208*11*8*33*75*9*09708356697052*772*94*72*032   occurrences of *scattered words*
11-33--9 and 772--72

These occurrences of scattered words are occurrences of previous words restricted to letters from subsets {1, 3, 9} and {7, 2} of alphabet A.

Remark

It is important that occurrences of scattered words are *only* those substrings of a text which are obtained by restricting occurrences of segments to all letters from a subset of an alphabet.

Notation: #w is the number of occurrences of a word or a scattered word in a text.

Definition 1

A word w is *complete* if and only if for every extension u of w, #u < .5#w.
(Meaning: w is not mainly "part of" any bigger word; it stands for itself.)

Definition 2

A proper *scattered sub-word* v of a *word* w is an *identifier* of w if and only if v occurs in w only once and #w > .5#v.
(Meaning: most occurrences of v are parts of w.)

Definition 3

A word w is *interesting* in a text T if and only if w is *complete* and contains at least one *identifier* v.

Theorem (easy)

The number of all occurrences of all words w that are complete in a text T of length n is bounded by $n*\log_2(n)$.

Three "informal" properties of *interesting words*

1. There are few or no interesting words in texts that are created from individual letters by "random" processes.

2. Editing texts by the "copy and paste" method usually creates *interesting words*.

3. To edit out interesting words from a text seems to be very tedious.

Problems

Thank you