EFFICIENT STRING MATCHING

Science fair

Fredrik Koppelow Eliassen

THE PAPER

Programming Techniques
Glenn Manacher Editor

Efficient String Matching: An Aid to Bibliographic Search

Alfred V. Aho and Margaret J. Corasick
Bell Laboratories

This paper describes a simple, efficient algorithm to locate all occurrences of any of a finite number of keywords in a string of text. The algorithm consists of constructing a finite state pattern matching machine from the keywords and then using the pattern matching machine to process the text string in a single pass. Construction of the pattern matching machine takes time proportional to the sum of the lengths of the keywords. The number of state transitions made by the pattern matching machine in processing the text string is independent of the number of keywords. The algorithm has been used to improve the speed of a library bibliographic search program by a factor of 5 to 10.

1. Introduction

In many information retrieva
cations it is necessary to be able
all occurrences of user-specifiec
phrases in text. This paper des
algorithm to locate all occurre
number of keywords and phra
string.

The approach should be fam
with finite automata. The algorit
In the first part we construct fro
finite state pattern matching mac
we apply the text string as input
machine. The machine signals
match for a keyword.

Using finite state machines ir
cations is not new [4, 8, 17], bu
frequently shunned by programr
for this reluctance on the part
due to the complexity of progra
algorithms for constructing finit
expressions [3, 10, 15], particula
techniques are needed [2, 14]. T
efficient finite state pattern ma
constructed quickly and simply f
regular expressions, namely thos
of keywords. Our approach co
Knuth-Morris-Pratt algorithm [1
state machines.

Perhaps the most interesting
the amount of improvement th
gives over more conventional ap
finite state pattern matching algo
graphic search program. The pu

THE AHO-CORASICK ALGORITHM

- Quickly locate occurrences of user-specified keywords from an arbitrary text string.
- 2 parts:
  - Construction of the finite state pattern machine
  - Locate keywords from string input

THE RELEVANCE OF THE ALGORITHM TO THE COURSE

- Search