Algorithmen und Datenstrukturen in der Stringverarbeitung

Vorlesung SS 2004

Hans Leiß, CIS

Voraussetzungen

Proseminar „Formale Sprachen und Automaten“

Literatur

D.Gusfield: Algorithms on Strings, Trees, Sequences. Cambridge UP, 1997

G.A.Stephen: String Searching Algorithms. World Scientific, 1994.

M. Crochemore: Text Algorithms. Oxford University Press, 1994

M. Crochemore, W.Rytter: Jewels of Stringology. World Scientific, 2002.

G. Navarro, M. Raffinot: Flexible Pattern Matching in Strings. Cambridge UP, 2002.

Inhalt (Plan)

Datenstrukturen:

(i) Tries, Suffix-Bäume (ggf. Suffix-Arrays),

(ii) Wortgraphen (Directed Acyclic Word Graphs)

(iii) ESP-Baum (Edit Sensitive Parse tree)

Indexierung und Suchprobleme:

(i) Erstellung des Suffixbaums eines Texts in linearer Zeit

(ii) Exakte/approximative Substringsuche (Matching)

(iii) Stringabstände und Matching mit Subwortverschiebung

Gleichheit und Minimierung (Multiset Discrimination):

(i) Schwaches Sortieren (ohne vorgeg. $<$)

(ii) Gleichheit von Listen und Bäumen bis auf Äquivalenz

(iii) Minimierung von Graphen in $O(n \log n)$

(iv) Minimierung von Transduktoren mit Suffixbäumen (evtl.)

Algorithmen zur Textkomprimierung:

(i) Lempel-Ziv Stringkompression (compress/gzip in Unix)

(ii) Suche in komprimierten Texten.

Algorithmen und ihre Abhängigkeiten

Indexierung:

(i) Vorlesung: Suffix-Array: einfache Konstruktion, weniger Speicherplatz als Suffixbaum, $O(|t| \log |t|)$.

(ii) Vorlesung: On-line-Konstruktion des Suffixbaums (Ukkonen) mit $O(|t|)$-Beweis.

(iii) Vorlesung: (evtl.) Suffixbaum mehrer Texte (Lazy in SML)

(iv) Seminar: Ukkonen-Verfahren in C^{++} und SML, ggf. Weiner-Verfahren implementieren und mit Ukkonen vergleichen

Suchalgorithmen:

(i) Voraussetzungen: endliche Automaten (exakte Suche), Stringabstände (Hamming, Edit, L_1), Dynamisches Programmieren, ESP-Baum

(ii) Vorlesung: Knuth-Morris-Pratt, Bit-Paralles Shift-Or (Erklärung mit Beweis); Stringabstände und Suche mit Verschiebungen (Theorie)

(iii) Seminar: ESP-Baum-Konstruktion, String-Vergleich mit Verschiebung (?), Shift-Or; ggf. KMP-Automat mit mlton Kompilieren und mit grep vergleichen

Kompression:

(i) Voraussetzungen: LZ-Trie, endliche Automaten

(ii) Vorlesung: LZ-78, LZ-77 (oder LZW), LZ-automat

(iii) Seminar: C-Implementierung gzip ansehen?

Gleichheit und Minimierung:

(i) Voraussetzung:

(ii) Vorlesung: Diskriminierung für Multimengen (Theorie + SML), Graphminimierung (Theorie)

(iii) Seminar: Minimierung von Dags, Minimierung von Graphen(?)