Multiple String-Suche: Verfahren von Aho-Corasick

Der Vortrag von Elena Atanasova behandelte das Thema der Multiplen String-Suche. Zu Beginn stellte sie das Prinzip des „Pattern Matching“ vor. Dabei sucht man zum Beispiel in einem String oder einem Korpus nach Mustern oder Schlüsselwörtern. Diese Vorgehensweise wird heute in vielen Bereichen der Informatik immer wichtiger. Als einfachste Variante des „Pattern Matching“ stellte sie den Naiven Ansatz vor. In diesem Naiven Ansatz sucht man nach einem Muster **x** der Länge **m** im Text **t** der Länge **n**. Die Laufzeit bei diesem Ansatz beträgt O(n*m). Ein verbesserter Ansatz, ist der Knurt-Morris-Pratt-Algorithmus. Bei diesem Algorithmus werden Informationen über bereits gewonnene Erkenntnisse bei der Zeichensuche abgespeichert. Bei dem Nativen Ansatz wird dies nicht getan und so muss das Muster immer wieder neu durchlaufen. Die gespeicherten Informationen werden in einer Sprungtabelle gespeichert, auf diese Weise wird wiederholtes durchlaufen der Suche verhindert. Auch werden dadurch wiederholte Vergleiche vermieden und die Laufzeit ist zudem deutlich geringer, nämlich nur O(n+m). N ist die Länge eines Textes und m ist die Gesamtzahl der Zeichen aller Schlüsselwörter. Die bisherig erwähnten Verfahren behandelten nur die Suche nach einzelnen Strings. Wenn man mehrere verschiedene Wörter sucht würde die Laufzeit O(m+k*n) betragen, wobei k die Anzahl der gesuchten Wörter ist. Dabei ist für jedes Wort eine individuelle Suche erforderlich. Der Text wird also jedes Mal erneut durchlaufen.

Um dieses Problem effizienter zu lösen, zieht man einen endlichen Automaten hinzu. Der Aho-Corasick-Algorithmus wurde von den beiden kanadischen Informatikern Alfred V. Aho und Margeret J. Corasick entwickelt. Mit Hilfe dieses Algorithmus kann man mehrere Schlüsselwörter gleichzeitig suchen. Dabei wird ein deterministischer endlicher Automat konstruiert. Zur Beschreibung wird das 6-Tupel $(Q,\sum,g,f,o,q_0)$ verwendet. Q bezeichnet eine endliche Menge von Zuständen, $\sum$ ein endliches Eingabealphabet, g die Übergangsfunktion „goto“, f eine Fehlerfunktion, o eine Ausgabefunktion und q_0 den Startzustand.

Die Übergangsfunktion g repräsentiert die Schlüsselwörter in einer Baumstruktur. Der Baum unterliegt folgenden Regeln. Jeder Baum hat eine

Wurzel q_0. Die Kanten tragen Zeichen des Alphabetes. Je 2 ausgehende Kanten eines Knotens haben unterschiedliche Zeichen. Die Suche nach Wörtern erfolgt durch Ablaufen der Kanten. Jeder Weg von der Wurzel zu einem Blatt entspricht einem Muster.

Die Konstruktion des Tries beginnt bei der Wurzel und führt entlang der Zeichen aus den verschiedenen Pattern. Falls der Weg vor Abschluss des Patterns zu Ende ist, fügt man fehlende Zeichen hinzu. Am Endknoten des Weges speichert man einen Identifier i von P_i.

Wenn man also den Baum mit einem Pattern durchgeht, und bei einem Identifier endet, so ist das Pattern enthalten. Wenn der Weg vor Ende des Patterns endet, so ist das Pattern nicht enthalten.

Die Fehlerfunktion f definiert die Fehler-Links. Ein solcher Link verweist von einem Zustandsknoten v auf einen Knoten w im Baum. Fehler-Links der Knoten der Tiefe 1 führen immer zur Wurzel zurück.

Die Ausgabefunktion o gibt zu jedem Zustand eine Menge der Schlüsselwörter an, die in diesem Zustand gefunden wurden.

Die Laufzeit des Algorithmus beträgt nur noch $O(m+n+k)$. Der Aho-Corasick-Algorithmus findet in vielen Bereichen Verwendung. Zum Beispiel in der Bildverarbeitung, oder bei der Erkennung von Virenmustern. Auch in der Bioinformatik, bei der Untersuchung der DNA wird der Algorithmus verwendet.