Gliederung

Implementation of KSEARCH and SANS Algorithms in the genometools-environment

Meike Bruns, Florian Markowsky,
Michael Spohn

University of Hamburg
Center for Bioinformatics
Bundesstraße 43
20146 Hamburg, Germany

http://www.zbh.uni-hamburg.de
info@zbh.uni-hamburg.de

Outline

Concept

Algorithms
- KSEARCH
- SANS

Implementation
- Our implementation
- Comparison to developers implementation

Sequenzvergleiche

- Die Bestimmung von Ähnlichkeiten zwischen Sequenzen ist eine häufige Aufgabe bei der Sequenzanalyse
- Exakte Sequenzvergleiche sind oft zu zeitaufwändig

Sequenzvergleiche

- Alignmentbasierte Methoden
  - SSEARCH
    - ermittelt optimale lokale Alignments mittels Smith-Waterman-Algorithmus
  - FASTA
    - verbindet HotSpots zu approximativen Alignments
  - BLAST
    - erweitert Hits zu Maximum Segment Pairs
- Ranking mittels Feature-Vektoren
  - USEARCH
    - Flo?
  - KSEARCH
    - bewertet Sequenzpaare aufgrund gemeinsamer k-mere
  - SANS - bewertet Sequenzpaare aufgrund potentieller gemeinsamer Substrings

Scoring - Funktion

$$KSEARCH(query, database) = \sum_{u \in \mathcal{A}^k} F(query, u) \cdot F(database, u)$$

Beispiel

query = BITQAPS
database = PQAPSQAP

kmer	query	database	Produkt
APS	1	1	1
BIT	1	0	0
ITQ	1	0	0
PQA	0	1	0
PSQ	0	1	0
QAP	1	2	2
SQA	0	1	0
TQA	1	0	0

Implementierungsansatz und Analyse

- Koskinen & Holm
  - Berechnung des KSEARCH-Scores mittels Suffixarrays:
    - Laufzeit: $O(|TXT| * |TXT_Q|)$ + lineare Konstruktion der Suffixarrays
    - Speicherplatzbedarf:
- Unsere Implementierung
  - Berechnung der KSEARCH-Scores mittels k-mer-Profilen
    - Laufzeit: $O(|TXT| + |TXT_Q| + (m+n) \cdot |\mathcal{A}^k| + AnzahlQprots \cdot AnzahlSprots \cdot 1)$???
    - Speicherplatzbedarf: $O(|AnzahlQprots| \cdot \mathcal{A}^k + AnzahlQprots \cdot AnzahlSprots)$

Idee

- Substrings von Sequenzen, die im Suffixarray nebeneinander liegen, haben mit hoher Wahrscheinlichkeit gemeinsame Präfixe
- SANS-Scores nutzen das aus, indem der Score aus den Positionen der Sequenzen im Suffixarray abgeleitet werden.

$$SANS(qprot, sprot) = \sum_{s=START[qprot]}^{START[qprot+1]-1} \sum_{i=-w}^{+w} id(sprot, SAP[ISA_{mapped}[s] + i])$$

Mit der Identitätsfunktion

$$id(a, b) = \begin{cases} 1, & if\ a = b \\ 0, & if\ a \neq b \end{cases}$$

Implementierungsansatz und Analyse

- Koskinen & Holm
  - Ermitteln der Positionen durch Mergen der Suffixarrays
    - Laufzeit: $O(|TXT| + |TXT_Q|)$ + lineare Konstruktion der Suffixarrays
    - Speicherplatzbedarf:
- Unsere Implementierung
  - Ermitteln der Positionen durch for-Schleifen
    - Laufzeit: $O(|TXT| + |TXT_Q|)$ + Konstruktion der Suffixarrays
    - Speicherplatzbedarf: $O(m \cdot n)$ + Platz für Suffixarrays

KSEARCHER

genometools