SVEUČILIŠTE U ZAGREBU

FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

BIOINFORMATIKA 1

Navarrov algoritam za približno uspoređivanje teksta

Fran Medved, Adam Vuković

Voditelj: Krešimir Križanović

Zagreb, svibanj, 2024.

**Sadržaj**

[1.](file:///C:\Users\franm\Downloads\Navarrov_algoritam_za_približno_uspoređivanje_teksta_2020%20(1).docx#_gjdgxs) Uvod 1

[2.](file:///C:\Users\franm\Downloads\Navarrov_algoritam_za_približno_uspoređivanje_teksta_2020%20(1).docx#_30j0zll) Algoritam2

[3.](file:///C:\Users\franm\Downloads\Navarrov_algoritam_za_približno_uspoređivanje_teksta_2020%20(1).docx#_1fob9te) Usporedba 5

[4.](file:///C:\Users\franm\Downloads\Navarrov_algoritam_za_približno_uspoređivanje_teksta_2020%20(1).docx#_3znysh7) Literatura 6

# Uvod

Približno uspoređivanje teksta jedan je od temeljnih zadataka bioinformatike. To je tehnika koja se koristi za pronalaženje podudarnosti između tekstualnih nizova kada dopuštamo određene razlike, kao što su umetanja, brisanja ili zamjene znakova. Klasična implementacija ove tehnike koristila je dinamičko programiranje te je zahtijevala vremensku složenost od O(mn), gdje je n duljina teksta, a m duljina uzorka koji tražimo u tekstu.

Danas se koristi pretraživanje uzoraka u hipertekstu. U takvoj implementaciji, tekst je prikazan kao graf gdje čvorovi sadržavaju dio teksta, a bridovi označavaju iduće dijelove teksta koji mogu slijediti trenutni čvor. Amir i suradnici razvili su algoritam koji ima vremensku složenost od O(m(n log m + e)) i prostornu složenost od O(mn), gdje je e broj bridova.

U sklopu ovog projekta, naš zadatak je implementirati Navarrov algoritam. No, budući da nas zanima i poravnanje sekvence sa grafom, mi ćemo koristiti cijelu matricu, a ne samo čuvati redak po redak.

# Algoritam

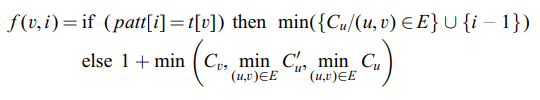
**Opis algoritma**

Prvo opišimo klasični algoritam. U ovom algoritmu stvaramo matricu *Ci, j* na način da je *Ci, 0 = C0, i = i* za sve *i*, te za sve ostale elemente na način:

*Ci, j = ako (a[i] = b[j]) onda Ci−1, j−1*

*inače 1 + min(Ci−1, j, Ci, j−1,Ci−1, j−1),*

No naš algoritam ne radi na tekstu, već na grafu. Mi algoritam započinjemo inicijalizirajući matricu, te postavljamo prvi red matrice na 0. Ako sa C’v označimo vrijednost pojedinog čvora trenutne iteracije, a sa Cv vrijednost čvora prošle iteracije, tada je formula za ispunjavanje matrice sljedeća:



**Primjer 1**

A diagram of a diagram

Description automatically generated

Pokazat ćemo rad algoritma na jednostavnoj linearnoj topologiji te pokazati poravnanje sa sekvencom **CGAT**.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **T** | **C** | **G** | **G** | **C** | **A** | **G** | **C** |
|  | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **G** | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 |
| **C** | 2 | 1 | 0 | 1 | 1 | 1 | 1 | 1 |
| **A** | 3 | 2 | 1 | 1 | 2 | 1 | 2 | 2 |
| **T** | 3 | 3 | 2 | 2 | 2 | 2 | 2 | 3 |

Kako bi dobili poravnanje, uzmemo najmanju vrijednost u zadnjem retku matrice. U ovom slučaju, ako uzmemo prvu ćeliju s brojem 2, backtrackingom dobijemo sljedeće poravnanje:

CG--

CGAT

**A diagram of a diagram

Description automatically generatedPrimjer 2**

# 

U ovoj topologiji, svaki čvor (osim prva dva) imaju 2 prethodnika. Rezultanta matrica je sljedeća:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | A | C | G | T | A | G | C | T | G | G |
|  | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| A | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 |
| C | 1 | 1 | 1 | 1 | 1 | 2 | 0 | 1 | 1 | 1 |
| G | 2 | 2 | 1 | 2 | 2 | 1 | 1 | 2 | 0 | 0 |
| T | 3 | 3 | 2 | 2 | 2 | 2 | 2 | 1 | 1 | 1 |
| A | 4 | 4 | 3 | 3 | 2 | 3 | 3 | 2 | 2 | 2 |

Backtrackingom dobijemo:

A-G-A

AAGGA

# Vremenska usporedba

Uspoređujemo vremensku izvedbu s rezultatima dobivenim u radu Bit-parallel sequence-to-graph alignment (<https://academic.oup.com/bioinformatics/article/35/19/3599/5372677?login=false>)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Cellbycell | Bitvector | Navarro |
| Linear | 50s | 2,4s | 213s |
| Snp | 75s | 4,1s | 267s |
| twopath | 217s | 23,5s | 559s |

# Literatura

[1] Mikko Rautiainen, Veli Makinen, Tobias Marschall; *Bioinformatics*, Volume 35, Issue 19, 1 October 2019; Bitt-paralleel sequence-to-graph alignment https://academic.oup.com/bioinformatics/article/35/19/3599/5372677

[2] Gonzalo Navarro; *Theoretical Computer Science*, Volume 237, Issues1-2, 28 April 2000; <https://www.sciencedirect.com/science/article/pii/S0304397599003333>