IN[34]120 Søketeknologi - Introduction & strings

2023-09-08 10:15 @ Chill

Gruppelærer: Oliver (Ruste Jahren), oliverrj@ifi.uio.no

(Join denne mentien!!)

- Praktisk/administrativ info
- Inverted indeces
- Posting lists
- Python refresher
- Språktek 101
- Suffix arrays
- Tries
- Oblighjelp

Grl: Oliver Ruste Jahren

- → 5. år på ifi
- → FUI (Maps, LI:ST)
- → Grl i søketek i fjor òg 😎
- → Bsc språktek, msc prosa

Søketeknologiens kjerne

- → Språktek
- → Prosa
- → (Robotikk?)

Emnets tematikk

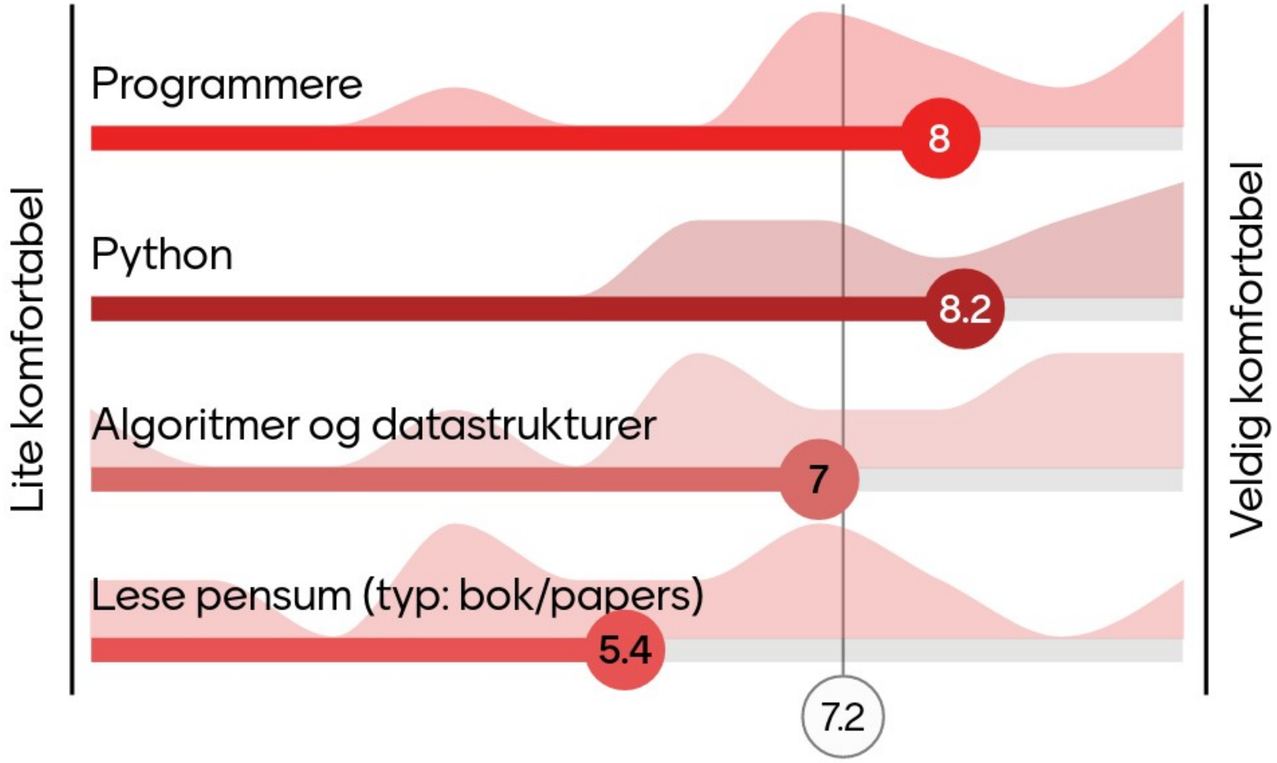
- Data
- Algoritmer
- Datastrukturer
- Maskinlæring / classification
- (Og mer)

Hva slags forventninger har du til søketek?

- Lære hvordan relevant informasjon hentes fra dokumenter
- Lære om de indre funksjonene til søkemotorer
- at foreleseren snakker høyere....
- tricky
- Ta over verden
- Vanskelig
- Tungt

There's no correct answer!

Hvor komfortabel er du med hver ting? (anonymt)



Hvor enig er du i hvert utsagn?

Assorterte stykker praktisk/administrativ info

(Viktig å vite)

Github

- Emnets kjerne
- Pensum
- Obliger
- Gruppetime-materiale etc. Alt annet enn opptak
- https://github.com/aohrn/in3120-2023

Mattermost

- (Open source Slack)
- Team
- Kanaler
- ((*Husk å vise hvordan man joiner kanaler*))
- Info

Gruppe 1-mattermost

- Vår egen kanal 😎
- Bli med
- Uklart formål (det blir bra)
- https://mattermost.uio.no/ifi-in3120/channels/group-1

</praktisk info>

(</x> betyr at x er ferdig, det er en SGML-greie)

Lecture recap

Altså fra forrige-forrige onsdag, 2023-08-24

Husker noen noe??

Ting som ble husket fra forelesningen.
Gjerne stikkord:

Inverted index	Praktisk info, invertert indeks, postinglister	Inverted index, posting list, dictionary
Masse "it depends"	Term	ikke fixed arrays

There's no correct answer!

Språktek-begreper 101

- Korpus
- Document
- Term
- Type
- Posting
- Query
- Boolean
- Retrieval
- "Boolean retrieval"

Skillet mellom strukturerte og ustrukturerte data

(Det ligger litt i navnet)

Strukturerte data

- → Definert format
- → Ofte et eksisterende formål
- → JSON, XML, UML(?)
- → Trivielt anvendbar

Ustrukturerte data

- "Vi må lage vår egen struktur"
- Rå tekst
- Data fra ikke-foremålstjenelige kilder
- Må behandles for å kunne brukes til noe
- Parsing

Parsing m/venner

- → Tokenisering
- → Stemming / Lemmatisering
- → Stoppord

Tokenisering

- → "token" = "ord", for det meste
- → Basic: splitte på mellomrom
- → Fancy: "United Kingdom" er 1 token

Lemmatisering

- Samle forskjellige former av ord til stamme
- "bok", "bøker" og "boka" blir alle "bok"
- Bevare semantikk

Stoppord

- "a", "the", "her"
- Betyr ikke noe
- Mange av dem -> dyrt å behandle
- Ignorer!

Inverted index

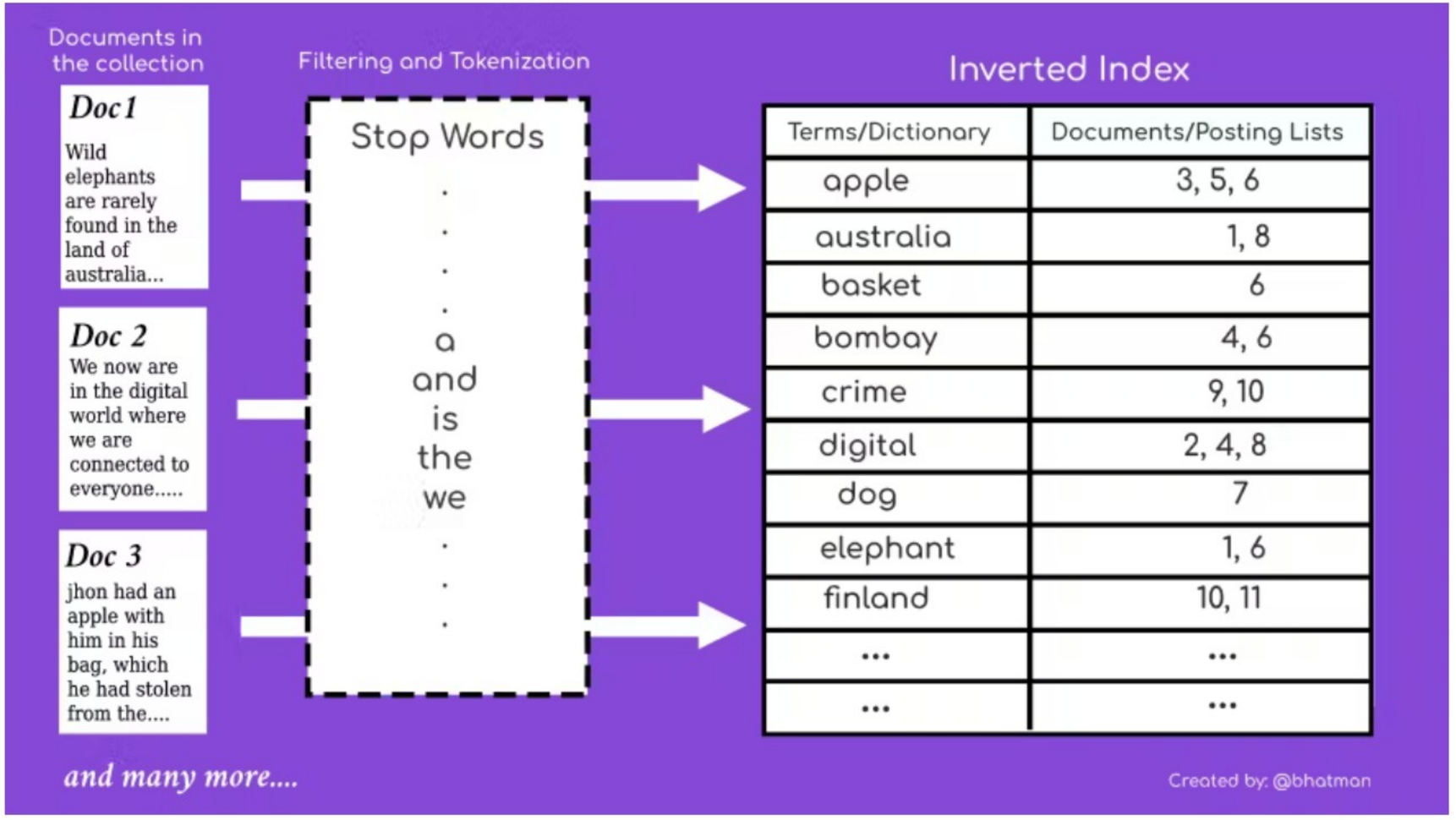
- → Mapping: term -> posting list
- → Som registeret i ei bok
- → 1/2 Oblig A (2023-09-15 (1 uke til))

Posting list

- En mengde dokumenter
- Alle dokumentene inneholder minst ett ord som er likt
- "her er alle dokumentene med 'Zeus'"

Posting lists forts.

- → Effektivitet: OOP?
- → Optimalisering: tall
- → 1 - 4 - 6 - 9
- → NB: Må være sortert



Visualisering: inverted index m/posting lists

Primitivt søk

- → Boolsk relevan -> ingen ranking
- → Ingen tolerans (må ha 100% lik stavemåte)

Hvorfor må posting lists være sortert?

- Enkelt se dokumenter som matcher søket ✗
- Hvis man finner en ID som er høyere enn dokumentet man sjekker, kan man avslutte tidlig ✗
- For å kunne sammenligne dem effektivt ✗
- Enklere søk, merging, bookeeping ✗
- for å merge svar effektivt ✗

The correct answer is: Slik at man kan gjøre effektive operasjoner på dem

Operasjoner på posting lists

- → Union (det som er i begge)
- → Intersection (det som kun er i den ene)
- → 2/2 Oblig A

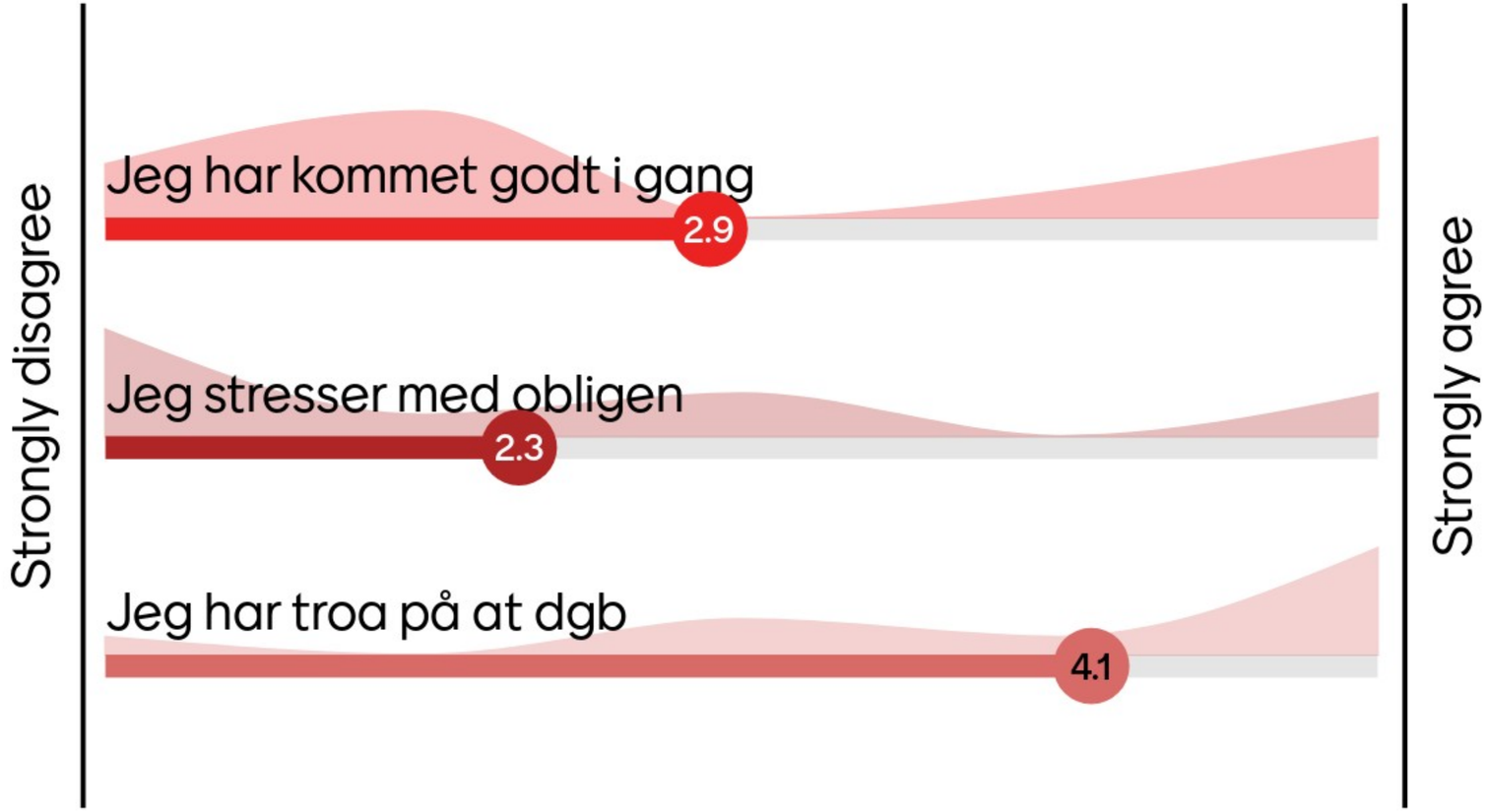
Oblig A

- Inverted index
- Postings-merger: union
- Postings-merger: intersection
- PM: Konstant minne
- Generators
- (2023-09-15)

Generators i Python

- Alternativ til å returnere verdier
- Se https://www.uio.no/studier/emner/matnat/ifi/IN3120/h20/material-for-group-sessions/advanced_python.pdf
- (Kreves for alle obliger)

Status for oblig A (frist om 1 uke)



"Adhere to the API"

Nøkkelen til å mestre prekoden. Les kommentarer og følg speccen

Strings (lecture 2)

Algorithms and data strucutres

Suffix arrays

- → Data structure for search
- → Find matching terms from suffixes
- → Sorted lexigraphically - why?

Suffix Array Example

Given String: banana

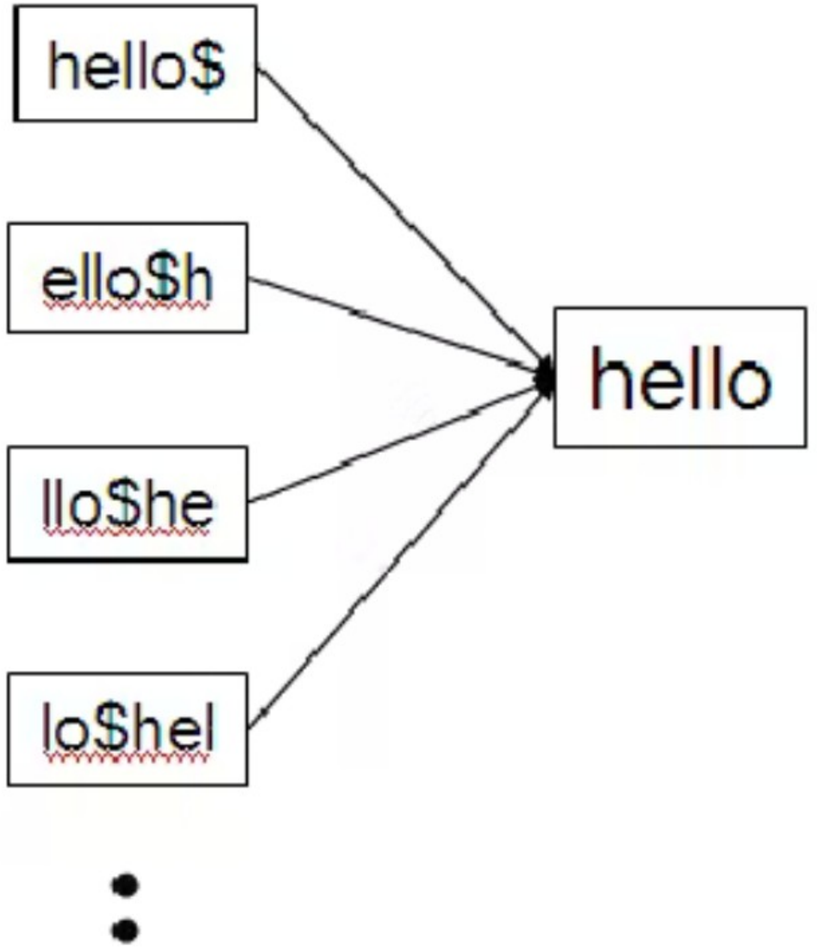
Suffixes		Sorted Suffixes
0 banana		5 a
1 anana	Sort the Suffixes	3 ana
2 nana	---------->	1 anana
3 ana	alphabetically	0 banana
4 na		4 na
5 a		2 nana

Suffix array: {5, 3, 1, 0, 4, 2}

Visualisering av suffix et basic suffix array (Oblig B)

Permuterm indeces

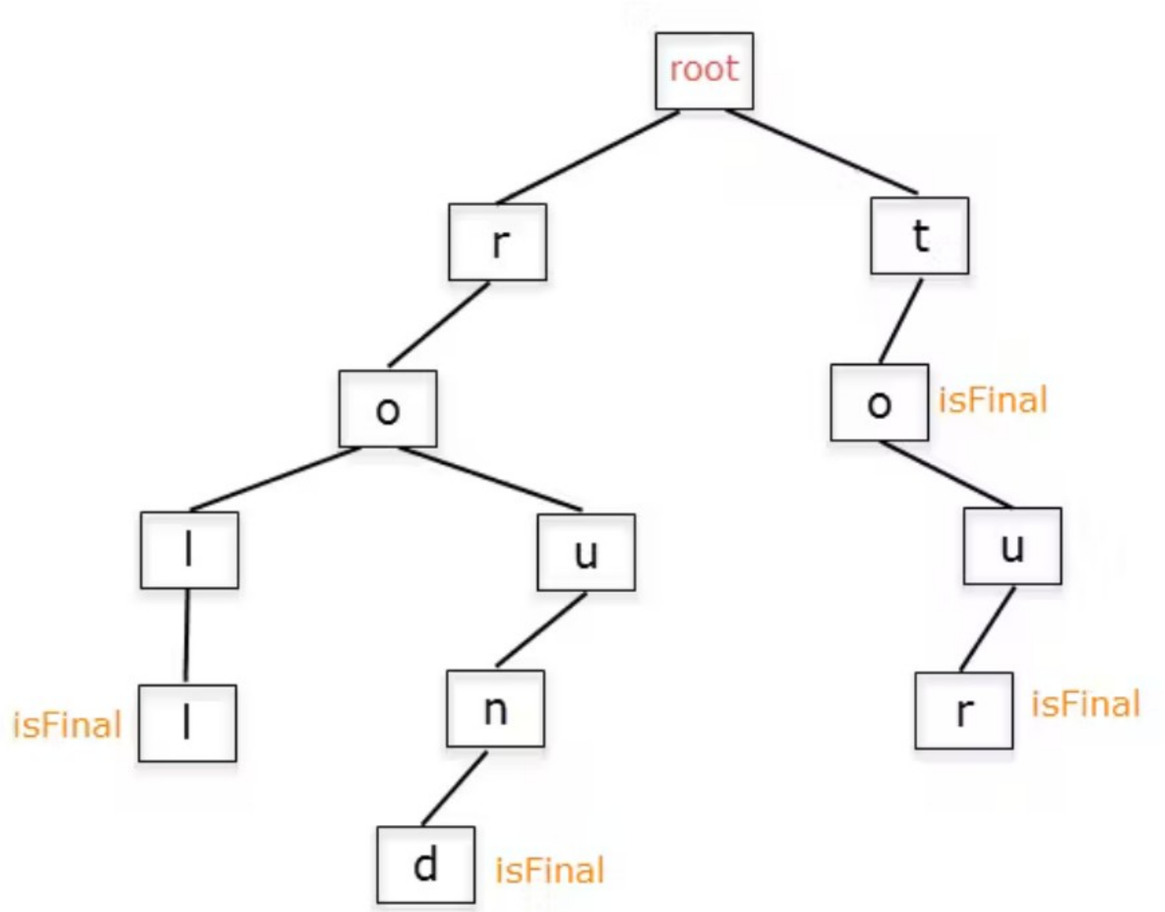
- "permutasjoner av typene"-index
- Støtter wildcard queries, e.g. /.ake/
- lsh litt samme use case som suffix arrays
- Lagrer typene fra korpuset "rotert"



Permuterm-index-struktur for termen "hello"

Tries

- Data structure - prefix tree
- Finn ut om en streng inngår i et korpus (raskt)
- Oblig B
- Mer om dette neste uke



Visualisering av trie (oblig B) (neste gang)

Spørsmål? 

Resten av tiden (*til 12*): Oblighjelp/innstallasjon

Neste gang: strengealgoritmer og oblighjelp

Spørsmål? Mattermost, mail, brevdue: oliverrj@ifi.uio.no