IN[34]120 - Søketeknologi 🖥️📚🔍

2024-11-19 - Siste gruppetime! 🕯️✨

Tema: Eksamensprep II

Agenda:
- Repetisjon: crawling + PageRank
- Utvalgte eksamensoppgaver (2)
- Originale øvningsoppgaver (6)

ti. 12. nov.	14:15–16:00	Eksamensforberedelser I.	OJD, Datastue Chill	O. R. Jahren	Først noen tips for innspurten, så løser vi gamle eksamensoppgaver i fellesskap.
ti. 19. nov.	14:15–16:00	Eksamensforberedelser II.	OJD, Datastue Chill	O. R. Jahren	Vi løser gamle eksamensoppgaver i fellesskap.

Eksamen fredag 29.11. Gruppe 1 morgen.

Siste gruppetime.

IN4120: Science fair 📚

- Var i går 🙀
- Registrert i Devilry 🤝

E-1 LF ✅

- → Kommer trolig ut torsdag
- → Kan få før i DMs - ta kontakt

Nytt pensum i år:

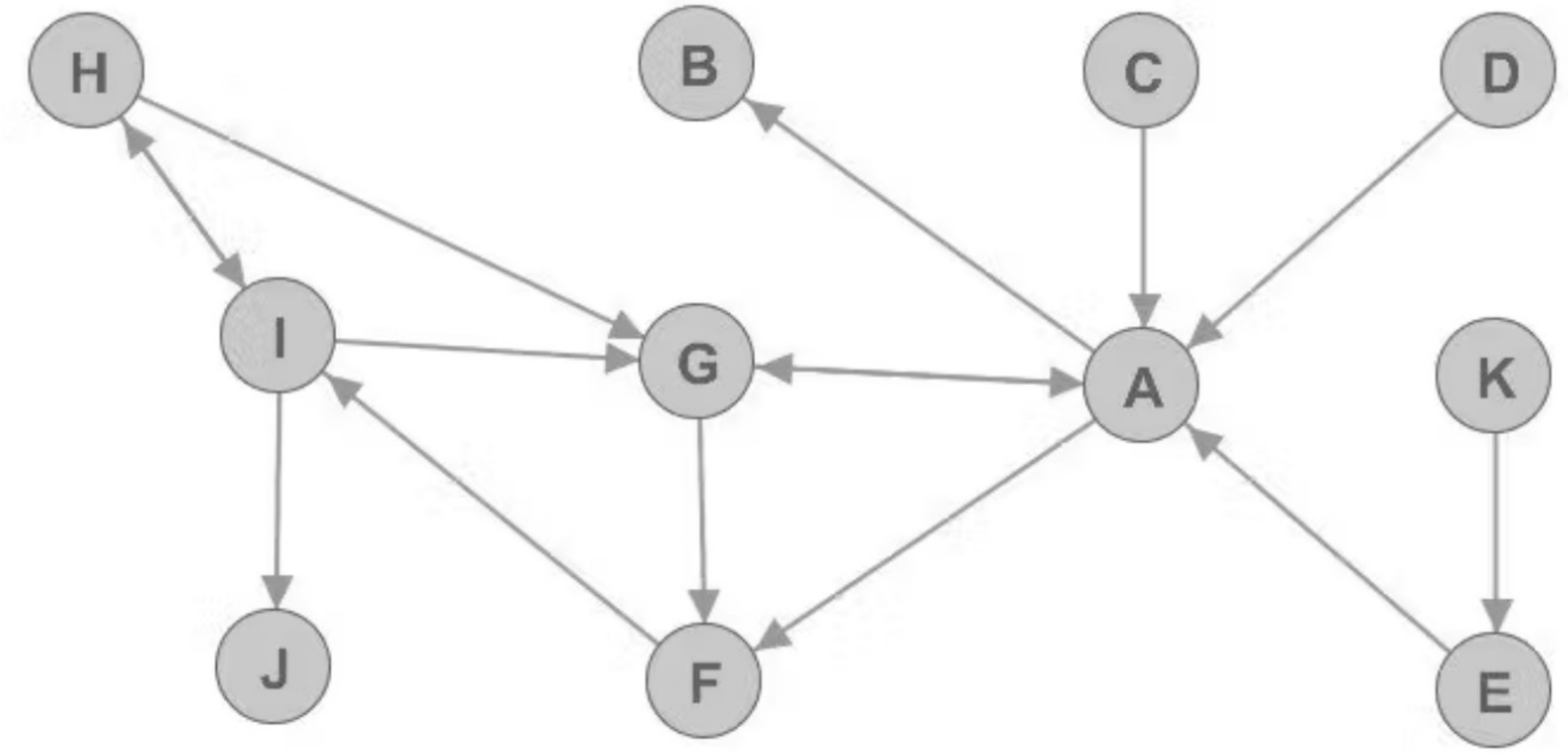
- Intet stort.
- En del ny prekode

Crawling og PageRank

Hvordan henger disse sammen?

Crawling 

- Knyttet til: Indeksering
- Finnes ikke en liste over alle nettsider
- Finne ut hvilket innhold som finnes på nettet
- "*Oppdage internettet*"



Når man starter i E, hvordan finne ut hva som finnes?

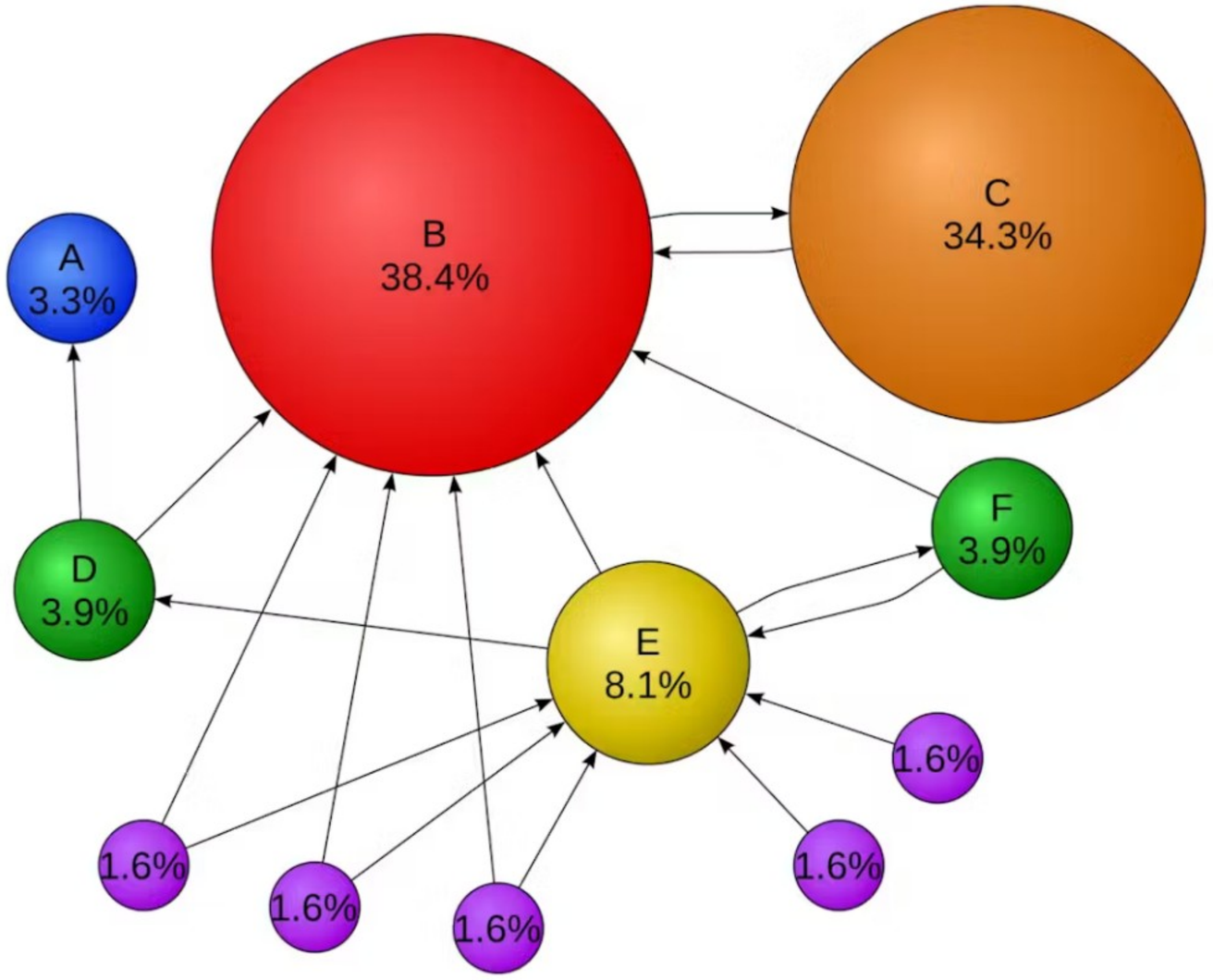
Internett som graf.

PageRank 📄🏆

- → Knyttet til: Ranking
- → Algoritme for å finne static score
- → Antar man har en ferdig graf
- → ^hvor får man grafen fra?

WEB SEARCH

- Basic principle: Query -> sorted list of ranked web pages
- The web is a directed graph of sites. Links are edges and pages are nodes.
  - Graphs allow us to use maths, logic and algorithms!



PAGERANK

- Ranking algorithm for web pages
  - Invented by and named for Larry Page, Google co-founder. Photo on the left.
    - Ofc also a good pun. *See if you can figure out why.*
  - Main principle:
    - If several sites {a, b, c} are linking to site x, site x is probably a good site and should be ranked highly(? Grammar).
      - (…Obviously?)
      - PageRank is a big deal because of finding a way of caluclating this.

Begrepet "traversal" 

- Knyttet til: Grafteori
- Generelt uttrykk
- *Å traversere grafen*
- Både crawling og PageRank

Futher reading

- Repo: slides/web-search-pierre.pdf
- Boka: Kap. 20
- Praktisk: https://uio-in3110.github.io/lectures/web/web.html
- Demo: https://github.com/Eckhoff42/Google-at-home

Bli gruppelærer her neste år 🤝

- Lærerikt 📚
- Hyggelig ❗
- Anbefales 🔥💯

Samme som forrige uke

Format på arbeidet 

1. spørsmål vises på tavla
2. diskuter med gruppe(4ish) 5-10 min
3. oppsummerer spørsmålet i fellesskap
4. gjenta

orgnaiser i grupper

a) [10p] Using an inverted index, discuss at least two ways to support fielded search. What are their advantages and drawbacks?

"Oppgave 6" - her avsluttet vi sist

a) E.g., see Section 6.1 here. The “fat dictionary” approach, or the “fat postings” approach. Should ideally provide examples/discussion related to, e.g., query performance, index size, operational flexibility, and more: Given the amount of points that are up for grabs, a high-scoring answer is expected to have a well-rounded discussion and display some creativity on applying different facets of what they’ve learnt in the course.

"Oppgave 6" LF

b) [10p] Some fields might intuitively be more important than others. For example, if your query matches the content of the *title* field for a document that might be better than a match in the *footnotes* field. Discuss how you could design a relevance function that takes this into account.

Oppgave 1

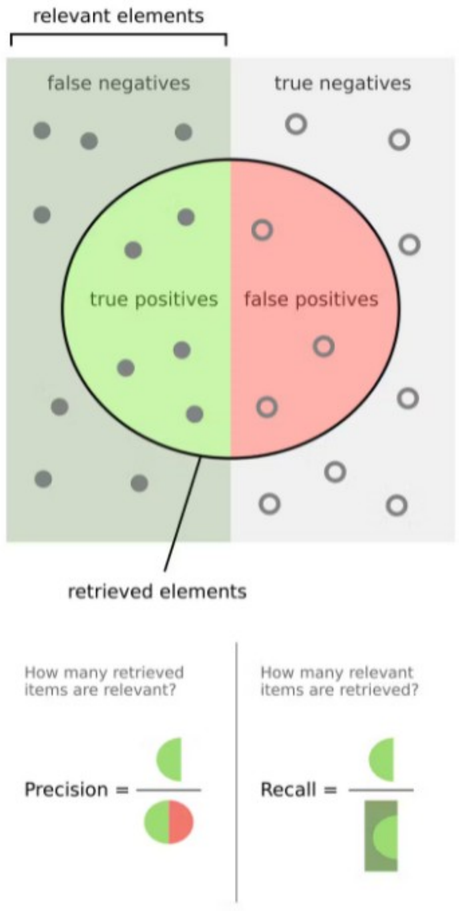
b) E.g., see Section 6.1.1 here, or consider tiering your index based on field importance. Should ideally provide examples/discussion of how to apply the field weights in practice, and how we might best determine what the weight values should be. See also comment above.

Oppgave 1 LF

a) [4%] Define, precisely, the two metrics precision and recall. Give examples of situations where you'd clearly want to prioritize one over the other.

Oppgave 2

Oppgave 2 LF


relevant elements
false negatives
true negatives
true positives
false positives
retrieved elements
How many retrieved items are relevant?
Precision =
How many relevant items are retrieved?
Recall =

Oppgave 2 LF

Oppgave 3

Hvis man kun skulle hatt dynamisk eller static score - hva burde man ha? Hvorfor?

Hvorfor bruker man vanligvis begge?

Oppgave 3 LF

Mitt svar: Dynamisk score siden den i størst grad tar hensyn til queryen.

Static score er uavhengig av queryen og ville mistet nyanser i innholdet.

Man bruker ofte begge fordi static score også er viktig for praktiske applikasjoner.

Oppgave 4

Er det viktig å behandle query-termene i samme rekkfølge som brukeren skrev dem? Hvorfor?

Finnes det situasjoner hvor rekkefølgen er viktig/ikke?

Oppgave 4 LF

Vurder: "Er høner haner ?"

Hvorvidt det er viktig å ta rekkefølgen i betraktning kommer an på applikasjonen.

Ved bruk av den BOW-modellen er rekkefølgen uvesentlig (ref NaiveBayes)

Oppgave 5

Forklar hva MapReduce er.

Anta du snakker til en andreårs ifi-bachelorstudent.

Oppgave 5 LF

MapReduce er en metode for å (1) få oversikt over data, og (2) prosessere denne slik at man får lagret den.
Det er en metode som kan kjøre på flere maskiner samtidig.
Dette er gunstig siden man kan gjøre flere ting samtidig og man har backup hvis en maskin dør.

Oppgave 6

Ranking: Forklar forskjellen mellom dynamisk og statisk score.

Anta du snakker til en andreårs ifi-bachelorstudent.

Oppgave 6 LF

Når du søker på noe tenker man at resultatene burde matche max det man søker på. Dynamic score ordner dette.

Men man har også noe som heter static score, som er uavhengig av hvor *mye* innholdet matcher og som handler om hvor BRA nettsiden er.

I praksis vil man ha en kombo.

Oppgave 7

Forklar hva edit distance er og hva man kan bruke det til. Gi eksempler på hvor man kan bruke det.

Anta du snakker til en andreårs ifi-bachelorstudent.

Oppgave 7 LF

Edit distance er en måte å tallfeste hvor like/ulike 2 strenger er. To like strenger har edit distance 0.

Man kan f.x. bruke det til å foreslå riktig ord når noen skriver feil. "Riktig ord" er et ord i ordlisten vår med lav edit distance mot det de skrev.

Oppgave 8

Forklar hvorfor det er gunstig å kunne gjøre ting *i minnet* fremfor *på disk*.

Anta du snakker til en andreårs ifi-bachelorstudent.

Oppgave 8 LF

Disk er tregt(!!!), minnet er raskt.

Minne-aksess: 100 nanosecs
Lese fra SSD-disk: 150k nanosecs.

1500x raskere!

Sauce:
https://gist.github.com/hellerbarde/2843375

Lykke til med eksamen!

- Takk for semesteret 🙏
- Query eleison 🕯️🙏✨🖥️📚🔍✨🙏

Closing remarks?

Break until 15:15 😄