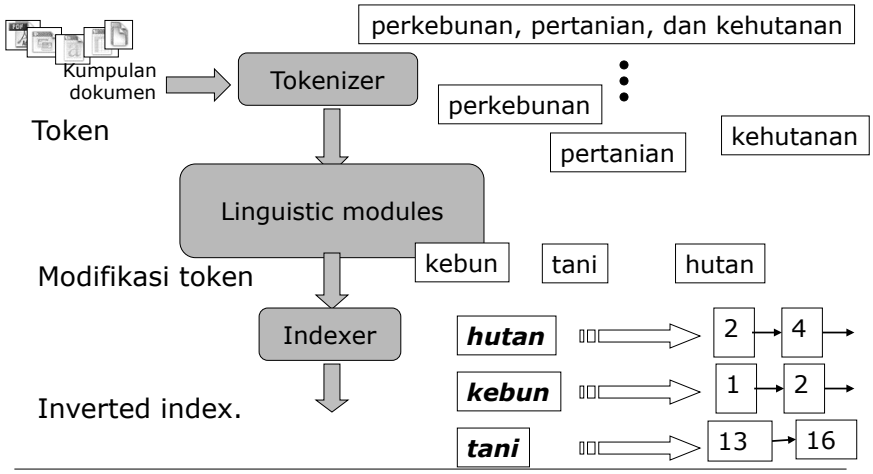
KOM341
Temu Kembali Informasi

KULIAH #3
- Inverted Index

??

- Apa persamaan pokok bahasan antara Rijbergen Ch.2 dengan Manning Ch.2?
- Apa perbedaannya?

Inverted index construction



Indexing

- Pengindeksan secara manual (oleh manusia)
  - Menentukan kata kunci dari suatu dokumen berdasarkan perbendaharaan kata yang ada (controlled vocabulary)
  - Oleh ahli di bidangnya
  - Lama dan mahal
- Pengindeksan secara otomatis
  - Program komputer untuk menentukan kata atau frase tertentu dari teks pada dokumen
  - Prosesnya cepat

Indexing

	Manual	Automatic
Controlled vocabulary	Catalogization	Categorization
Free text	Indexing	Search engine

Tahap Pengindeksan Otomatis

- Perhatikan struktur dokumen (id, tanggal, author, title, text, dsb)
- Tokenisasi
- Buang stopwords
- [proses pemotongan imbuhan (stemming)]
- Pembobotan kata
- Pembuatan indeks

1. Struktur Dokumen

- Tergantung jenis dokumen
- Format umum : SGML
- Perlu pemisahan antar elemen dokumen?
- Bagian mana saja dari dokumen yang akan di-indeks?

2. Tokenisasi

- Kata atau konsep?
  - Kata sederhana (tunggal)
  - Frase
  - Konsep → Thesaurus
- Teknik: segmentasi, memilah
- Tokenisasi :: Frase
  - Contoh frase : tusuk jarum
  - Banyak menggunakan metode statistika
  - Berdasarkan frekuensi kemunculan
  - POS-tagger

3. Stopwords

- Kata buangan, daftar kata umum yang mempunyai fungsi tapi tidak mempunyai arti
- Contoh : dan, atau, yang
- Dari pengalaman, frekuensinya sangat banyak
- Dibuang, untuk efisiensi
- Timbul masalah, misalkan :
  - Yang Mulia
  - Yang Maha Kuasa
  - Ekspresi DAN bernilai benar jika kedua operand bernilai benar

4. Stemming

- Proses pembuangan prefiks dan sufiks (secara morfologi) dari suatu kata berimbuhan menjadi kata dasar.
- Contoh: menyelesaikan → selesai (stem)
- Stemming dilakukan atas dasar asumsi bahwa kata-kata yang memiliki stem yang sama memiliki makna yang serupa pula.
- Banyak riset menunjukkan bahwa stemming tidak mempengaruhi kinerja temu kembali secara nyata.

Tujuan Stemming

- Efisiensi, mengurangi jumlah kata-kata unik dalam indeks sehingga mengurangi kebutuhan ruang penyimpanan untuk indeks dan mempercepat proses pencarian.
- Efektivitas, meningkatkan dokumen yang ditemu-kembalikan (recall) dengan mengurangi varian kata menjadi bentuk kata dasarnya (stem).

Masalah Stemming

- Umum dilakukan dalam IR walaupun hasilnya banyak bermasalah.
- Contoh:
  - kadaluarsa →ngadaluarsa
  - mencapai →capa
- Stemming akan menghilangkan variasi morfologi.
  - **kabar** : berkabar, mengabarkan, terkabar, perkabaran, pengabaran

Teknik Stemming

Teknik stemming dapat dikategorikan menjadi tiga jenis:

- Berdasarkan aturan sesuai bahasa tertentu,
- Berdasarkan kemunculan bersama,
- Berdasarkan kamus.

Porter Stemmer

- Stemmer untuk bahasa Inggris yang terdiri dari beberapa aturan, dikembangkan oleh Martin Porter.
- Dasar algoritme Porter berdasarkan pemotongan awalan (prefix) dan akhiran (suffix):
  - Penghitungan ukuran kata
  - Aturan pemotongan.

Ukuran kata

- Setiap kata/bagian kata dapat memiliki salah satu dari empat bentuk:
  - KVKV ... K
  - KVKV ... V
  - VKVK ... K
  - VKVK ... V
- Bentuk umum:
  - [K]VKVK ... [V] atau
  - [K] $(VK)^m$ [V]

  dimana ***m*** melambangkan ukuran kata

Ukuran kata

- Contoh:
  - ”makan” dan ”bentuk” : m=2
  - ”presentasi” dan ”dimanakah” : m=4
- Fungsi penghitung ukuran kata digunakan untuk mencegah stemming menghasilkan stem yang terlalu pendek (overstemming).
- Diasumsikan minimal stem berukuran dua, kecuali jika kata berukuran kurang dari dua.

Aturan pemotongan

- Aturan: P1 (kondisi) S1 → P2 S2
- Artinya jika kata memiliki prefiks P1 dan sufiks S1, dan bagian kata setelah P1 dan sebelum S1 memenuhi kondisi atau aturan yang diberikan, maka P1 dan S1 akan diganti menjadi P2 dan S2.
- Kondisi dapat menggunakan operator AND, OR, atau NOT untuk menyatakan aturan yang kompleks.

Contoh

(m>1) *wan* →

- berarti S1 adalah *wan*, dan S2 adalah null (kata kosong).
- Contoh: kata ”dermawan” dipotong menjadi ”derma” karena ”derma” berukuran 2 (m>1).

Program Stemmer

- Berbagai implementasi Porter Stemmer untuk bahasa Inggris:
  - http://www.tartarus.org/~martin/PorterStemmer/
- Untuk bahasa Indonesia, lihat :
  - UI : Neil Siregar (1995)
  - UI : Adriani & Nazief (1996)
  - IPB : Ahmad Ridha (2002), Julio Adisantoso (2009)
- Bantuan kamus.

Teknik stemming

- Periksa semua kemungkinan bentuk kata:

  prefiks1+prefiks2+KATADASAR+sufiks3+sufiks2+sufiks1

- Lakukan pemotongan berurutan : prefiks1, prefiks2, sufiks1, sufiks2, sufiks3 (kalau ada), dan KATADASAR.
- Setiap tahap pemotongan diikuti dengan pemeriksaan di kamus kata dasar). Jika ada maka proses dinyatakan selesai.
- Contoh : kata mempermainkannya :
- Jika sampai pada pemotongan sufiks3 masih belum ada di kamus, maka dilakukan proses kombinasi.

Kombinasi kata

- ~~Kata Dasar~~
- ~~Kata Dasar + Akhiran 3~~
- ~~Kata Dasar + Akhiran 3 + Akhiran 2~~
- ~~Kata Dasar + Akhiran 3 + Akhiran 2 + Akhiran 1~~
- Awalan 1 + Awalan 2 +Kata Dasar
- Awalan 1 + Awalan 2 +Kata Dasar + Akhiran 3
- Awalan 1 + Awalan 2 +Kata Dasar + Akhiran 3 + Akhiran 2
- ~~Awalan 1 + Awalan 2 +Kata Dasar + Akhiran 3 + Akhiran 1~~
- Awalan 2 + Kata Dasar
- Awalan 2 + Kata Dasar + Akhiran 3
- Awalan 2 + Kata Dasar + Akhiran 3 + Akhiran 2
- ~~Awalan 2 + Kata Dasar + Akhiran 3 + Akhiran 2 + Akhiran 1~~

Awalan me-

- tetap, jika huruf pertama kata dasar adalah l, m, n, q, r, atau w. Contoh: *me-* + luluh → *me*luluh, *me-* + makan → *me*makan.
- *me-* → *mem-*, jika huruf pertama kata dasar adalah b, f, **p***, atau v. Contoh: *me-* + baca → *mem*baca, *me-* + pukul → *mem*ukul*, *me-* + vonis → *mem*vonis, *me-* + fasilitas + i → *mem*fasilitasi.
- *me-* → *men-*, jika huruf pertama kata dasar adalah c, d, j, atau **t***. Contoh: *me-* + datang → *men*datang, *me-* + tiup → *men*iup*.
- *me-* → *meng-*, jika huruf pertama kata dasar adalah huruf vokal, **k***, g, h. Contoh: *me-* + kikis → *meng*ikis*, *me-* + gotong → *meng*gotong, *me-* + hias → *meng*hias.
- *me-* → *menge-*, jika kata dasar hanya satu suku kata. Contoh: *me-* + bom → *menge*bom, *me-* + tik → *menge*tik, *me-* + klik → *menge*klik.
- *me-* → *meny-*, jika huruf pertama adalah **s***. Contoh: *me-* + sapu → *meny*apu*.

Sifat khusus

- Dilebur jika huruf kedua kata dasar adalah huruf vokal. Contoh: *me-* + *ti*pu → *meni*pu, *me-* + *sa*pu → *menya*pu, *me-* + *ki*ra → *mengi*ra.
- Tidak dilebur jika huruf kedua kata dasar adalah huruf konsonan. Contoh: *me-* + *kl*arifikasi → *mengkl*arifikasi.
- Tidak dilebur jika kata dasar merupakan kata asing yang belum diserap secara sempurna. Contoh: *me-* + *k*onversi → *mengk*onversi.

Masalah pada stemming

- Understemming
  - jumlah kata/imbuhan yang dipotong terlalu sedikit
  - Misal: ”pengorbanan” menjadi ”korbanan”
- Overstemming
  - jumlah kata/imbuhan yang dipotong terlalu banyak
  - Misal: ”mencapai” menjadi ”capa”

5. Pembobotan

- Perlunya suatu kata diberi bobot
  - Makin sering suatu kata muncul pada suatu dokumen, maka diduga semakin penting kata itu untuk dokumen tsb.
- Beberapa pendekatan:
  - tf
  - tf.idf
  - BM25
  - dsb.

Term frequency (tf)

- Frekuensi kemunculan suatu term ***t*** pada dokumen ***d*** ➔ $tf_{t,d}$
- Contoh:

term	d1	d2	d3	d4	d5
dari	20	100	10	22	10
database	15	0	0	0	12
dengan	12	40	12	14	24
informasi	8	30	0	0	18
komputer	10	35	0	0	0
struktur	0	24	6	10	10
yang	35	120	15	32	20

Term frequency (tf)

- Mana yang lebih memberikan informasi sebagai penciri dari suatu dokumen? (kasus pada dok-1)
  - Kata komputer yang muncul sebanyak 10 kali
  - Kata yang yang muncul sebanyak 35 kali
  - Kata dari yang muncul sebanyak 20 kali
- Kata yang muncul di semua dokumen dengan frekuensi yang besar. Apa akibatnya?

Document frequency (df)

- Banyaknya dokumen di dalam koleksi yang mengandung kata tertentu.
- Mana yang lebih informatif bagi suatu kata untuk mencirikan dokumen?
  - Seberapa jarang suatu kata muncul di seluruh dokumen?
  - Seberapa sering suatu kata muncul di seluruh dokumen?
- Contoh:

dari	database	dengan	informasi	komputer	struktur	yang
500	25	1500	34	12	48	750

Inverse document frequency (idf)

- Banyaknya dokumen dimana suatu term ***t*** muncul :

$$\frac{1}{df_t}$$

- Dikoreksi dengan banyaknya seluruh dokumen dalam koleksi (***N***), menjadi :

$$idf_t = \log\left(\frac{N}{df_t}\right)$$

Bobot tf.idf

- Hasil kali : $\boldsymbol{tf}_t$ x $\boldsymbol{idf}_t$
- Maka bobot setiap term ***t*** pada dokumen ***d*** adalah:

$$w_{t,d} = tf_t.\log\left(\frac{N}{df_t}\right)$$

- Kecenderungan nilai bobot:
  - Berbanding lurus dengan frekuensi kemunculan term ***t*** pada suatu dokumen ***d***.
  - Berbanding terbalik dengan banyaknya dokumen yang mengandung suatu term ***t***.

Beberapa alternatif tf.idf

Sublinear tf scaling

$$wf_{t,d} = \begin{cases} 1 + \log(tf_{t,d}) & tf_{t,d} > 0 \\ 0 & selainnya \end{cases}$$

sehingga

$$wf.idf_{t,d} = wf_{t,d}.idf_t$$

Beberapa alternatif tf.idf

Maximum tf normalization

Untuk setiap dokumen d, misalkan

$$tf_{\max}(d) = \max_{\tau \in d} tf_{\tau,d}$$

dimana τ adalah banyaknya setiap term dalam *d* maka

$$ntf_{t,d} = a + (1-a)\frac{tf_{t,d}}{tf_{\max}(d)}$$

Boolean

- Kemunculan suatu term ***t*** pada dokumen ***d*** → $b_{t,d}$ → [0,1]
- Contoh:

term	d1	d2	d3	d4	d5
dari	1	1	1	1	1
database	1	0	0	0	1
dengan	1	1	1	1	1
informasi	1	1	0	0	1
komputer	1	1	0	0	0
struktur	0	1	1	1	1
yang	1	1	1	1	1

Contoh kasus: N=10000 Hitung bobot tf.idf normal

- Daftar kata dan kemunculannya

	harga	saham	dunia	turun	investor	rugi
D1	3	10	3	0	5	9
D2	7	0	2	4	3	0
D3	1	4	7	6	2	6
D4	6	0	2	7	1	2

- Banyaknya dokumen yang mengandung kata

	harga	saham	dunia	turun	investor	rugi
#doc	100	4000	2000	500	1000	200