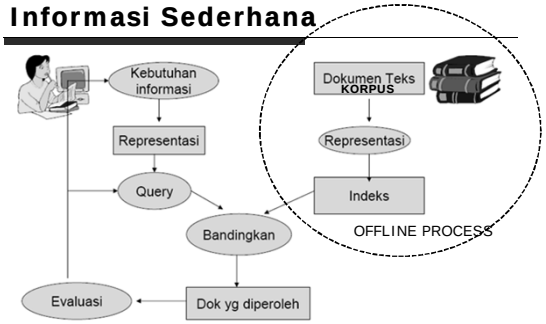
KOM341
Temu Kembali Informasi

KULIAH #2
- Pemrosesan Teks
- Pengantar bahasa PERL

Proses Perolehan Informasi Sederhana



Pengertian TEKS

- Teks ≈ Korpus ≈ Koleksi dokumen yang bisa dibaca oleh mesin
- Contoh:
  - Kumpulan artikel surat kabar yang diperoleh dari Internet
  - Kumpulan skripsi mahasiswa yang telah dikumpulkan secara digital oleh perpustakaan

Korpus

- Korpus adalah teks alami yang dipilih dengan cara tertentu.
- Masalah pada perancangan korpus
  - Ukuran
  - Jenis
  - Bahasa
- Media: teks, audio, video (multimedia)
- Isu pada korpus:
  - Tokenisasi pada korpus
  - Anotasi pada korpus

Contoh Korpus Free text

Sekurangnya 17 ribu ayam ras milik peternak di wilayah kabupaten Kotawaringin Timur (Kotim) , Kalimantan Tengah mati dan kuat dugaan akibat terserang virus avian influenza (AI) atau yang lagi ramai disebut penyakit flu burung. Kasubdin Produksi Peternakan Dinas Pertanian Kotim Drh. Mawardi di Sampit, Selasa mengatakan sebanyak 17 ribu ekor ayam ras yang mati diduga terserang flu burung itu sejak Desember 2003.

Dari hasil diagnosa Balai Penyelidikan dan Pengujian Veteriner (BPPV) regional V Banjar Baru Kalimantan Selatan yang diterima Disnak Kotim, Senin (26/1) menyebutkan ayam yang mati terserang panyakit itu hanya ada dua kemungkinan yaitu terserang virus AI dan VVND atau tetelo. "Namun kasus kematian masal ayas ras di Kotim kemungkinan besar akibat akibat serangan virus avian influenza yang bila menular kepada manusia namanya menjadi flu burung," ucapnya.

Contoh Korpus XML Format

```
<DOC>
<DOCNO>DOC01</DOCNO>
<TITLE>Flu Burung Menyerang Kalimantan Tengah</TITLE>
<AUTHOR>Ark, Ant</AUTHOR>
<DATE> 7 Februari 2003 </DATE>
<TEXT>
<P>Sekurangnya 17 ribu ayam ras milik peternak di wilayah kabupaten Kotawaringin Timur (Kotim) , Kalimantan Tengah mati dan kuat dugaan akibat terserang virus avian influenza (AI) atau yang lagi ramai disebut penyakit flu burung. Kasubdin Produksi Peternakan Dinas Pertanian Kotim Drh. Mawardi di Sampit, Selasa mengatakan sebanyak 17 ribu ekor ayam ras yang mati diduga terserang flu burung itu sejak Desember 2003.</p>

<P>Dari hasil diagnosa Balai Penyelidikan dan Pengujian Veteriner (BPPV) regional V Banjar Baru Kalimantan Selatan yang diterima Disnak Kotim, Senin (26/1) menyebutkan ayam yang mati terserang panyakit itu hanya ada dua kemungkinan yaitu terserang virus AI dan VVND atau tetelo. "Namun kasus kematian masal ayas ras di Kotim kemungkinan besar akibat akibat serangan virus avian influenza yang bila menular kepada manusia namanya menjadi flu burung," ucapnya.</P>
</TEXT>
</DOC>
```

Melacak Teks

- Operasi dasar dalam string matching
- Contoh:
  - Dapatkan semua baris yang dimulai dengan kata Flu.
  - Dapatkan semua baris yang dimulai dengan kata Huruf Besar.
  - Dapatkan semua baris yang memiliki kata terdiri dari huruf besar semua.
  - Hitung banyaknya kata Flu pada dokumen tersebut.
  - dsb.

Statistik Teks

- Jumlah Kata
  - Seberapa besar korpus yang ada (N)
- Jenis kata
  - Berapa jumlah kata yang unik?
  - Berapa besar perbendaharaan kata pada korpus?
- Token kata
  - Berapa jumlah kata pada korpus?
  - Berapa frekuensi dari setiap jenis kata?
  - Kata apa yang paling sering muncul pada korpus?

Prosedur Menghitung Frekuensi Kata

- Tokenisasi : mendapatkan kata
- Ubah menjadi huruf kecil
- Urutkan menurut abjad
- Hitung frekuensi kemunculan kata
- Urutkan menurut frekuensinya
- Hitung frekuensi dari frekuensi kemunculan kata

Fenomena Frekuensi Kata

- Sejumlah kata merupakan kata yang sangat umum (frekuensi sangat besar), misalnya “the”, “of”
- Kebanyakan kata sangat jarang muncul (frekuensi sangat kecil).
- Setengah dari kata-kata pada korpus hanya muncul sekali.

Contoh

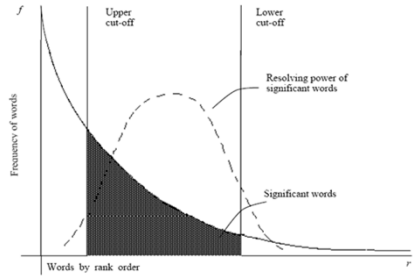
Kata	Frekuensi Kata (f)	Peringkat (r)	f * r
name	21	400	8400
comes	16	500	8000
group	13	600	7800
science	11	700	7700
family	10	800	8000
begin	9	900	8100
broke	4	2000	8000
seems	2	3000	6000
could	2	4000	8000

Hukum Zipf

- Menjelaskan adanya hubungan antara frekuensi dan urutan/rank (George Kingsley Zipf).
- Urutan/Rank:
  - hitung berapa kali kata muncul pada semua teks di dalam korpus (f).
  - urutkan sesuai dengan frekuensi kemunculan kata membentuk rank (r).
- Terdapat konstanta c sehingga f * r = c

Luhn's Ideas

kata-kata yang paling umum dan paling tidak umum adalah tidak signifikan untuk indexing



Stopwords

- STOPWORDS:
  - Terdapat kata yang merupakan bagian terbesar dari teks yang tidak perlu digunakan sebagai penciri dokumen.
  - Terdapat banyak ragam kata yang hanya muncul sedikit sekali di dalam suatu teks.
  - Contoh: to, in, form, yang, dan
- Kata-kata dengan frekuensi cukup (di bagian tengah) adalah yang paling baik digunakan sebagai penciri dokumen.

Tokenisasi

- Pengertian : suatu tahap pemrosesan di mana teks input dibagi menjadi unit-unit kecil yang disebut token, yang dapat berupa suatu kata, suatu angka, atau suatu tanda baca.
- Konsekuensinya:
  - Perlu mengenali unit secara otomatis
  - Apakah suatu kata itu?
  - Kalimat?
  - Paragraf?

Kata

- Karakter alfanumerik yang saling terhubung yang dipisahkan oleh whitespace.
- Whitespace: spasi, tab, newline
- Masalah:
  - B2B, amazon.com, Micro$oft
  - isn't, Jum'at
  - pro-aktif, out-of-date
  - tanda sambung pada akhir baris

Segmentasi kata

- Proses tokenisasi sederhana, tetapi tidak terlepas dari kesalahan.
- Contoh:
  - Kata majemuk: Jurusan Surabaya-Jakarta
  - Frase: tusuk jarum, keras kepala, sistem informasi
  - Nomor telpon (0251) 8356653 +62 251 8625584
- Menjadi topik dari ekstraksi informasi

Kalimat

- Satu atau lebih string kata yang diakhiri dengan suatu tanda berhenti sepenuhnya, tanda tanya atau tanda seru.
- Contoh:
  - Akhir dari baris.
  - Akhir dari suatu cerita!
  - Apakah kamu sudah punya pacar?
  - Ia sering mengunjungi friendster.com.
  - Dr. Iwan pergi ke Surabaya.
  - Dia mengatakan "Bohong!".

Batas kalimat

- Hipotesakan bahwa batas kalimat sesudah . ? !
- Pindahkan batas kalimat sesudah tanda petik, bukan setelah titik.
  - "Jangan ambil buku itu. Buku itu milik pak Budi. ", kata ibu kepada Ani.
- Jangan gunakan titik jika:
  - Sebelumnya adalah singkatan yg umum yg biasanya bukan akhir kalimat, tp biasanya diikuti oleh nama dengan huruf besar: Prof.
  - Didahului oleh singkatan yang umum dan tidak diikuti oleh kata dengan huruf besar: Jr.
- Jangan kenali sebagai batas jika ada ! atau ? yang diikuti oleh huruf kecil

Pemrosesan Teks Otomatis

- Menghitung kata pada teks
- Mengurutkan kata
- Menghitung berbagai nilai statistik kata
  - Input : berkas teks (bisa berupa korpus)
  - Output : daftar kata beserta frekuensinya
- Pertanyaannya:
  - Bagaimana program komputernya?
  - Bahasa pemrograman apa yang digunakan?

PERL

http://www.activestate.com/activeperl/downloads

- Practical Extraction and Report Language
- Dikembangkan oleh Larry Wall pada tahun 1987
- Mengembangkan suatu bahasa script yang lebih baik daripada Unix shell tetapi tidak serumit C.
- Berguna untuk memanipulasi teks yang tidak dapat dilakukan oleh instruksi baris unix

PERL

```
#!/usr/local/bin/perl
#Program untuk menulis kata 'hello'
print "Hello\n";
```

- Tulis pada suatu file, misalnya bernama 'halo.pl' (Unix), atau 'halo.pl ' (Windows).
- Untuk me-run pada Unix/Linux:
  `> perl halo.pl`
- Untuk me-run pada Windows:
  `> perl halo.pl`

PERL di web

```
#!"C:\apache\xampp\perl\bin\perl.exe"

print "Content-type: text/html\n\n";
print '<html>';
print '<head>';
print '<meta name="author" content="Kay Vogelgesang">';
print '<link href="/xampp/xampp.css" rel="stylesheet"
type="text/css">';
print '</head>';
print "<body> <p><h1>GCI with MiniPerl</h1>";
print  "CGI with MiniPerl is ready ...</body></html>";
```

Tulis pada suatu file, misalnya bernama 'test.cgi' dan simpan di folder cgi-bin.

PERL Jenis data

- Tiga jenis data dalam PERL:
  - Skalar
  - Array
  - Associative array atau hash
- Nama variabel
  - Nama variabel skalar dimulai dengan $ (mis. $dok)
  - Nama variabel array dimulai dengan @ (mis. @kata)
  - Nama variabel hash dimulai dengan % (mis. %tabel)
  - Nama variabel adalah case sensitive ($kelas ≠ $KELAS)

PERL
Skalar

- Angka
  - digits, desimal, eksponensial dll.
  - $nilai = 350;
  - $nilai = 3.50
- Strings
  - Diapit oleh tanda petik single / double;
  - Escape character dengan backslash
    - \n (newline) ; \t (tab) ;
    - \U (Uppercase) ; \L (Lower case)
  - print "\Uhalo\n"; → HALO
  - print "ha\Ulo\n"; → haLO

PERL
Operator

- Bilangan
  - Aritmatika : +, -, /, *, %
  - Assignment : =, +=, -=, ++, --
- String :
  - Concatenation : .
  - Repetition : x

PERL
Array

- Suatu array adalah suatu variabel yang berisi list
- Suatu array berisi nol atau lebih elemen. Tidak perlu ditentukan panjangnya seperti pemrograman lainnya.
- Contoh:

```
(1, 2, 3)
("yang", "dan", "untuk")
()
```


PERL
Array

```
@kata = ("yang", "dan", "untuk");
@x = (1, 2, 3);
@y = @x;    # assign nilai x ke y
@y = (@x 4 5);     # @y=(1 2 3 4 5)
$z = @y;  # $z bernilai 5 (panjang @y)
($z) = @y;  # $z = 1 (elemen pertama @y)
@prefix = $kata[0,1];  # ("yang", "dan")
```


PERL
Array

```
@array = ("aa", "bb", "cc", "dd");
$length = @array;  #4
print $#array;   #3
print $array[$#array];  #"dd"
print scalar(@array) ;  #4
($a, $b) = ("satu", "dua");
($satu, @b) = (1,2,3,4,5,6);
   # $satu = 1, @b=(2 3 4 5 6)
($a, $b) = ($b, $a);  #tukar nilainya
```


PERL
Mengubah isi array

Push

menambahkan list pada bagian akhir dari array

```
@a1 = ("aa", "bb", "cc", "dd");
@a2 = ("ee", "ff");
push @a1, @a2;
   #@a1=("aa","bb","cc","dd","ee","ff")
push @a2, "gg";
   #@a2=("ee","ff","gg")
```

PERL
Mengubah isi array

Pop
Membuang elemen terakhir dari list

```
@array = ("aa","bb","cc","dd");
$elemen = pop @array;
   # $elemen= "dd"
   # @array = ( "aa","bb","cc")
```


PERL
Associative Arrays

- Array yang memiliki indeks bukan berupa bilangan 0 sampai dengan n-1, melainkan string.
- Contoh:

```
%nilai=();
$nilai{"dan"}=200;
$nilai{"yang"}=150;
@kata=keys(%nilai);  #@kata=("dan", "yang")
@freq=values(%nilai);  #@freq=(200, 150)
```


PERL
Control Structures

- IF

```
if ($nilai > 60)
   { print "Lulus\n"; }
else
   { print "Tidak lulus\n"; }
```

- UNLESS

```
unless ($nilai > 60)
   { print "Tidak lulus\n"; }
```

- ELSIF

```
if ($nilai >= 80)
   { print "A\n"; }
elsif ($nilai >= 60)
   { print "B\n"; }
else
   { print "Tidak lulus\n"; }
```


PERL
Control Structures

- WHILE

```
$i = 10;
while ($i > 5) {
   $x = $i*$i;
   print "Kuadrat dari $i adalah $x\n";
   $i--;
}
```

- UNTIL

```
$i = 10;
until ($i <= 5) {
   $x = $i*$i;
   print "Kuadrat dari $i adalah $x\n";
   $i--;
}
```


PERL
Control Structures

- FOR

```
for ($i = 1; $i<=10; $i++)
{ print "$i\n"; }

@kata=("dan", "yang", "untuk");
for ($i=0 ; $i<=$#kata; $i++) {
   $prefix = $kata[$i];
   print "$prefix\n";
}
```

- FOREACH

```
foreach $prefix(@kata) {
   print "$prefix\n";
}
```


PERL
Associative Arrays

```
%nilai=();
$nilai{"dan"}=200;
$nilai{"yang"}=150;
foreach $i (keys %nilai)
   { print "$nilai{$i}\n"; }
while (($kata, $freq) = each (%nilai))
   { print "$kata : $freq\n"; }
foreach $i (sort keys %nilai)
   { print "$nilai{$i}\n"; }
```

PERL
Sorting

```
@kata = ("dan", "itu", "yang", "pada");
reverse(@nama);
@stopwords = sort(@kata);
```

- Sesuai alfabet: `sort {$a cmp $b} @list;`
- Sesuai numerik: `sort {$a <=> $b} @list;`
- Descending: `sort {$b <=> $a} @list;`
- Associative array

```
sort {$tabel{$b} <=> $tabel{$a} }
   (keys %tabel);
```


PERL
Input/Output

- Membuka dan menutup file

```
open(IN, "koleksi.txt");
open(OUT, ">hasil.txt");
open(OUT ">>hasil txt"); → append
close(OUT);
```

- Membuka dan menutup file

```
open(IN, "koleksi.txt");
open(OUT, ">hasil.txt");
while ($line = <IN>) {
   chop($line); # buang carriage return
   print OUT "$line\n";
}
close(IN);
close(OUT);
```


PERL
Regular Expression

- Ekspresi yang digunakan untuk menggambarkan pola dari suatu obyek.
- Sering digunakan dalam pemrosesan teks.
- Bahasa yang banyak menggunakan RE adalah PERL.
- Pemrograman menjadi lebih singkat dan mudah.

PERL
Regular Expression

\b	batas kata
\d	digit = [0-9]
\n	newline
\r	carriage return
\s	karakter white space
\t	tab
\w	karakter alfanumerik = [A-Za-z0-9]
^	awal dari string
$	akhir dari string

PERL
Regular Expression

.	karakter apapun
[bdkp]	karakter b, d, k dan p
[a-f]	karakter a sampai f
[^a-f]	semua karakter kecuali a sampai f
abc|def	string abc atau string def
*	nol atau beberapa kali
+	sekali atau beberapa kali
?	nol atau satu kali
qw()	("quote word") function qw(aa bb cc) → ("aa", "bb", "cc")

PERL
Regular Expression

/(ab)*(de)+/

abde
abbde
bde
abadde
ababde

/a{5}b{1,4}c{2}/

aaaaabcc
aaaaabcccc
aaaaabc
aaaabcc

PERL
Regular Expression

/c[ad]r/
car, cdr
cadr, caddddr

/c[ad]*r/
car, cdr, caaaadr, caaadaaar

/[a-g][t-z][0-9]*/
au125

PERL
Pengolahan String

- Pola string → **//**

```
$kata=~/tan/;
$kata=~/^tan/;
$kata=~/tan$/;
```

- Mengubah bagian string → **s///**

```
$kalimat=~s/minyak/BBM/;
$kalimat=~s/minyak/BBM/g;
$kalimat=~s/minyak/BBM/gi;
```

- Mengubah karakter → **tr///**

```
$kata=~tr/[A-Z]/[a-z]/;
```


PERL
Fungsi SPLIT

- Untuk tokenisasi, mendapatkan kata dari suatu kalimat dengan pemisah (delimiter) tertentu.
- split(///);
- Contoh

```
$kalimat= "Petani alami gagal panen";
@kata=split(/\s+/, $kalimat);
```

maka array

```
@kata=("Petani", "alami", "gagal", "panen")
```


PERL
Contoh Program

```
$kalimat="Harga minyak sekarang turun.
Sebelumnya harga minyak naik. Sekarang
harga minyak turun lagi";
$kalimat =~ tr/[A-Z]/[a-z]/;
@kata = split(/\s+/, $kalimat);
foreach $token(@kata) {
   print "$token\n";
}
```


PERL
Contoh Program

```
open (IN, "dokumen.txt") || die;
while($line = <IN>)
{
   chomp($line);  #buang carriage return
   $line =~ s/^\s*//;  #buang whitespace
   $line =~ s/\s*$//;
   reset @arraykata;
   @arraykata = split /\s+/, $line;
   #untuk setiap kata
   foreach $kata(@arraykata) {
      $kata =~ tr/[A-Z]/[a-z]/;
      $kata =~ s/[!.,()*]|\"//g;
      $freq{$kata}++;
   }
}
# Mencetak daftar kata, sort by frekuensi
foreach $key (sort{$freq{$b}<=>$freq{$a}}keys %freq){
   print "$key - $freq{$key}\n";
}
close(IN);
```