

Laporan Tugas 2

Teks dan Web Mining

Diajukan untuk melengkapi tugas Mata Kuliah Teks dan Web Mining

Oleh:

MUAMMAR ZIKRI AKSANA

1608107010045



JURUSAN INFORMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS SYIAH KUALA
DARUSSALAM, BANDA ACEH
MARET, 2019

I. Tahapan Penyelesaian

Pada tugas 2 membangun kamus untuk kedua kategori dengan n-grams kata dilakukan dengan tahapan :

1. Path dan file
 - didapatkan *path* folder dari file masing-masing kategori
 - lalu folder yang didapat diberikan perintah “ls”
 - untuk setiap list file yang ada dibaca dengan “cat” untuk setiap filenya
 - setiap file yang di-cat dihapus stopwordsnya dengan regex
 - lalu setiap bagian file (dipecah perkata agar kombinasi kata tidak terdiri dari kata yang tidak bermakna) diberikan ke method `Lingua::`
 - `EN::Bigram->ngram(n)` dimana `n` adalah n-grams kamus yang ingin dibangun
 - untuk setiap hasil n-grams dibaca dan diberikan kepada hash untuk dihitung count
 - setiap proses pembacaan file selesai dilanjutkan eliminasi
2. Eliminasi

Pada tahap eliminasi keys dari salah satu hash(struktur data untuk menyimpan word dari n-grams) dibaca, lalu dari keys yang didapat diberikan kepada *hash* kamus satunya , jika terdapat keys yang sama maka dilakukan *scenario* yang dijelaskan pada tugas (bab III).
3. Hasil eliminasi ditulis kedalam file

II. Pembacaan dan Pembagian File untuk n-grams

Didapatkan terlebih dahulu path dari folder yang berisi file yang ingin dibangun kamusnya.

Program yang dibuat untuk menyelesaikan tugas ini dijalankan dengan command line argument :

n-grams src threshold dst

dimana :

- n-grams : word grams kamus yang ingin dibangun
- src : path source , ditulis dengan format array dan json, dimana setiap index harus memiliki kategori dan path dimana path setiap kategori bisa terdiri dari banyak path
- threshold, besar threshold pada tahap eliminasi
- dst : hasil kamus disimpan

```
perl gKamus.pl 2
'[
  {"categori":"bola",
   "path":[
     "/media/mza/Pro/DataPro/data-clean/suaramerdeka/bola",
     "/media/mza/Pro/DataPro/data-clean/bola/bola"]},
  {"categori":"basket",
   "path":["/media/mza/Pro/DataPro/data-clean/bola/basket",
           "/media/mza/Pro/DataPro/data-clean/mainbasket/basket",
           "/media/mza/Pro/DataPro/data-clean/viva/basket",
           "/media/mza/Pro/DataPro/data-clean/juara/basket"]}
]'
40 /media/mza/Pro/DataPro/kamus/
```

a. N-grams dari file

Untuk mendapatkan word dengan *grams* yang ditentukan digunakan library `Lingua::EN::Bigram`. sebelum diambil word dengan *grams* tertentu file sumber dibersihkan dari stopwords. Setiap word yang didapat disimpan pada sebuah hash. Proses hashing dilakukan untuk setiap satu file. kembalian dari method `Lingua::EN::Bigram->ngram` adalah array di join setiap datanya dengan “\n”; lalu hasil tersebut diberikan ke fungsi insert struktur data.

```
for my $index (keys %hashSource){
  foreach my $path (@{$hashSource{$index}}){
    my $listFiles=`ls $path`;
    foreach (split "\n",$listFiles){
      my $content= getContent(`cat $path/$_`);
      $ngrams->text($content);
      # my @text=$ngrams->ngram(1);
      print "Start $start\n";
      print "insert at ".localtime."\n";
      insert($index,join "\n",$ngrams->ngram($ARGV[0]));
    }
  }
}
```

b. Struktur Data

Setiap content yang didapat di split dengan “\n” sehingga didapat array dengan string n-grams tertentu (sesuai n-grams proses b). proses insert :

- term diberikan ke hash jika hasil hash tidak “exists” maka inisialisasikan hash dengan memberikan nilai 1 dan *increment*-kan total untuk kamus kategori ini namun jika “exists” maka *increment*-kan nilai hash sekarang.

```
sub insert{
  foreach my $word (split "\n",$_[1]){
    # print "\tinsert :$word\n";
    if(exists $dataTheasaurus{$_[0]}{$word}){
      $dataTheasaurus{$_[0]}{$word}++;
    }else{
      $dataTheasaurus{$_[0]}{$word}=1;
      if(exists $totalData{$_[0]}){
        $totalData{$_[0]}++;
      }else{
        $totalData{$_[0]}=0;
      }
    }
  }
}
```

III. Eliminasi

Proses penghapusan duplikasi kata yang sama pada kamus dengan melakukan observasi eliminasi rasio untuk threshold 45% dan 50%. Eliminasi dilakukan dengan

- mengambil keys dari hash pada satu kamus
- setiap keys yang didapat di hashing ke struktur data hash kamus lainnya
- jika keys tersebut "exists" pada hash satunya, maka dilakukan eliminasi.

Eliminasi dilakukan dengan :

- jumlah count keys (word) dari setiap kamus dinormalisasi
- lalu dicari rasio dengan membandingkan kedua hasil normalisasi sebelumnya dengan penyebut adalah bilangan terbesar dari keduanya.
- rasio yang didapat dibandingkan dengan threshold,
- jika rasio lebih besar dari threshold maka keys harus dihapus dari kedua kamus karena keys tersebut common untuk kedua kamus.
- jika rasio lebih kecil dari threshold maka, keys (word) dengan nilai normalisasi terbesar dihapus dari kamusnya.

```
sub eliminasi2T{
  my $start=localtime;
  my @keys = keys %dataTheasaurus;
  foreach (keys %{$dataTheasaurus{$keys[0]}}){
    print "Start $start\n Eliminasi at ".localtime."\n";
    if(exists $dataTheasaurus{$keys[1]}{$_}){
      my $val1 = $dataTheasaurus{$keys[0]}{$_}/$totalData{$keys[0]};
      my $val2 = $dataTheasaurus{$keys[1]}{$_}/$totalData{$keys[1]};
      my $flag = $val1>$val2?$val1/$val2:$val2/$val1;
      if($flag > $threshold){
        delete $dataTheasaurus{$keys[0]}{$_};
        delete $dataTheasaurus{$keys[1]}{$_};
      }else{
        if($val1 > $val2){
          delete $dataTheasaurus{$keys[1]}{$_};
        }else{
          delete $dataTheasaurus{$keys[0]}{$_};
        }
      }
    }
  }
}
```

IV. Membangun Kamus

Setelah proses eliminasi selesai, file kamus dibangun berdasarkan data yang telah dihash dengan bentuk data json :

```
{
    "keys": [count,normalize],
    "keys2": [count,normalize].
    "nkeys": [count,normalize]
}
```

normalize dari tiap keys didapat dari perbandingan count dan jumlah keys dari tiap kamus (size hash).

file ditulis dengan format keys,count,count/total, total didapat dari proses insert.

```
mza@zx:/media/mza/Pro/DataPro/kamus/basket$ head -5 ngrams-1-40
{"yugianto":[1,2.68730517037515e-05],
"core":[3,8.06191551112544e-05],
"meneladani":[3,8.06191551112544e-05],
"bergenre":[4,0.000107492206815006],
"muggsy":[36,0.000967429861335053],
mza@zx:/media/mza/Pro/DataPro/kamus/basket$ tail -5 ngrams-1-40
"kebersamaa":[1,2.68730517037515e-05],
"mva":[1,2.68730517037515e-05],
"tuduh":[1,2.68730517037515e-05],
"dennies":[1,2.68730517037515e-05],
}mza@zx:/media/mza/Pro/DataPro/kamus/basket$ wc -l ngrams-1-40
20625 ngrams-1-40
```

diatas contoh hasil 5 line pertama dan 5 line terakhir dengan format json, dari file kamus 1-grams basket dengan total line 20625.

V. Hasil Akhir

Hasil akhir yang didapat dari tugas 2, adalah kamus dengan 1,2,3-grams dan masing-masing terdiri dari 2 file dengan 40% dan 50% threshold.

```
mza@zx:/media/mza/Pro/DataPro/kamus$ wc -l basket/*
  20625 basket/ngrams-1-40
  20625 basket/ngrams-1-50
 520858 basket/ngrams-2-40
 520858 basket/ngrams-2-50
1519722 basket/ngrams-3-40
1519722 basket/ngrams-3-50
4122410 total
mza@zx:/media/mza/Pro/DataPro/kamus$ wc -l bola/*
 22396 bola/ngrams-1-40
 22396 bola/ngrams-1-50
540341 bola/ngrams-2-40
540341 bola/ngrams-2-50
1568889 bola/ngrams-3-40
1568889 bola/ngrams-3-50
4263252 total
```

pada gambar diatas kamus dari tiap n-grams masing-masing kategori, pada output diatas tidak terdapat banyak perbedaan dari hasil yang didapat dari kedua threshold.