INDEKS *FILE* (**TEKS**) MENGGUNAKAN SWISH-E

by: M. SALSABILA JAMIL

SWISH-e adalah sebuah *tool* yang digunakan untuk meng-indeks teks dalam berbagai format, seperti PDF, html, txt, XML, PostScript, dll. Selain meng-indeks, *tool* ini juga dapat mencari dokumen berdasarkan *query* yang di-*input*-kan. *Tool* ini memiliki banyak fitur/kemampuan yang dimilikinya, contohnya:

- Dapat menentukan *stopword list* yang hendak digunakan
- Dapat menentukan direktori (berisi file) yang ingin di-indeks
- Dapat menentukan jenis file yang di-indeks (PDF?, XML?, email?, txt?, html?, dll)
- Bisa melakukan *stemming* (dalam beberapa bahasa)
- *Fast*
- Dapat meng-indeks berita dari suatu *website* (*crawling*)
- Dapat Mencari dokumen yang mengandung *query* yang dimasukkan
- Dan masih banyak lagi -> Baca Lebih Lanjut

Berikut dijelaskan cara menggunakan *tool* ini.

Langkah pertama yaitu ***DOWNLOAD FILE* BERISI TEKS** (disini penulis menggunakan TEKS BERITA)

1. Penulis membuat skrip untuk mengunduh berita pada website:
    - KOMPAS: `src/spider/spider_kompas.go`
    - TEMPO: `src/spider/spider_tempo.go`
    - ANTARA:
        - `src/spider/spider_antara.go`
        - `src/spider/spider_antara_oto.go`
2. Tema-tema berita yang di-*download*:
    - *Entertainment*/Hiburan
    - *Sport*/Olahraga
    - Tekno
    - Otomotif
3. Bagian-bagian *HTML* yang diambil:
    - *Title*/Judul
    - Tanggal Terbit
    - *Tag* (Opsional)
    - *Content*/Isi
4. Berita yang terkumpul Dapat dilihat pada *folder*:
    - `kompas/`
    - `tempo/`
    - `antara/`
5. Selanjutnya dilakukan proses ***PREPROCESSING***:
    - Menghapus `\n\r`
    - Menghapus tanda baca yang dianggap ***TIDAK PENTING**

    - Mengubah *spasi* lebih dari satu menjadi satu *spasi*
    - *Lowercase every letter*
    - Hasil disimpan pada folder:
        - `ALL/`
        - Secara keseluruhan berjumlah: ***~55K***

Langkah kedua, ***STEMMING***:

1. Pengertian: Mengubah kata ke bentuk dasarnya, contoh:
   _ berjalan -> jalan _ memancing -> pancing * menggunakan -> guna
2. Tujuan:
   Sebisa mungkin mengurangi kata yang nantinya akan di-indeks, sehingga jumlah memori yang digunakan semakin sedikit
3. Proses *STEMMING* menggunakan pustaka/modul yang dikembangkan oleh RadhiFadlillah dalam bahasa GO, dimana RadhiFadlillah mengambil referensi dari andylibrian yang mengembangkannya dalam bahasa PHP.
4. Pustaka (kopi sebahagian dari README.md RadhiFadlillah)

   Algoritma

   1. Algoritma Nazief dan Adriani
   2. Asian J. 2007. ***Effective Techniques for Indonesian Text Retrieval***. PhD thesis School of Computer Science and Information Technology RMIT University Australia. (PDF dan Amazon)
   3. Arifin, A.Z., I.P.A.K. Mahendra dan H.T. Ciptaningtyas. 2009. ***Enhanced Confix Stripping Stemmer and Ants Algorithm for Classifying News Document in Indonesian Language***, Proceeding of International Conference on Information & Communication Technology and Systems (ICTS). (PDF)
   4. A. D. Tahitoe, D. Purwitasari. 2010. ***Implementasi Modifikasi Enhanced Confix Stripping Stemmer Untuk Bahasa Indonesia dengan Metode Corpus Based Stemming***, Institut Teknologi Sepuluh Nopember (ITS) – Surabaya, 60111, Indonesia. (PDF)
   5. Tambahan aturan *stemming* dari kontributor Sastrawi.

   Kamus Kata Dasar

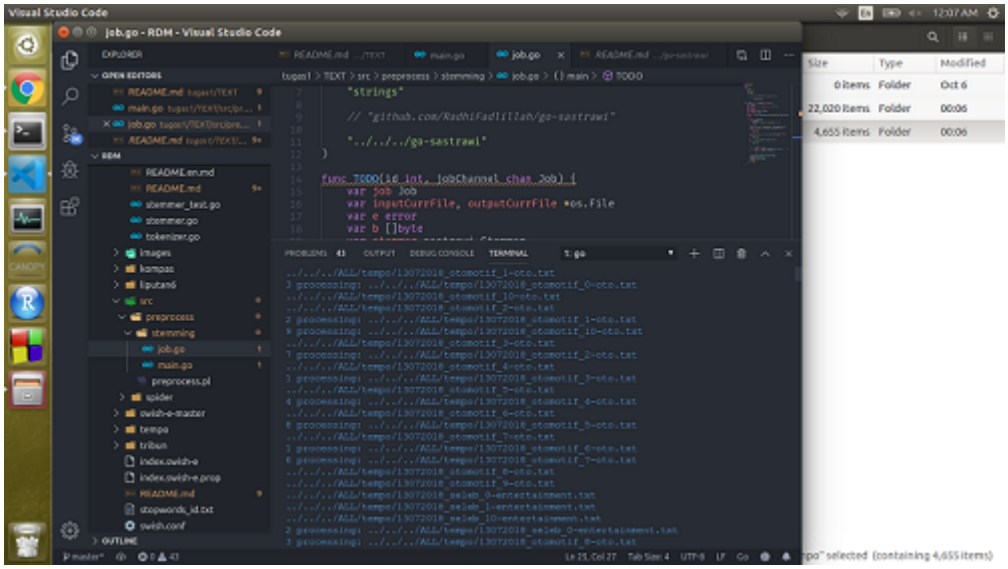
   Proses stemming oleh Sastrawi sangat bergantung pada kamus kata dasar. Sastrawi menggunakan kamus kata dasar dari kateglo.com dengan sedikit perubahan.

Lisensi

Sebagaimana Sastrawi untuk PHP, Go-Sastrawi untuk GO, projek ini juga disebarkan dengan lisensi MIT. Untuk lisensi kamus kata dasar dari Kateglo adalah CC-BY-NC-SA 3.0.

5. Proses *STEMMING* dilakukan menggunakan skrip yang terdapat di folder `src/preprocess/stemming/main.go`

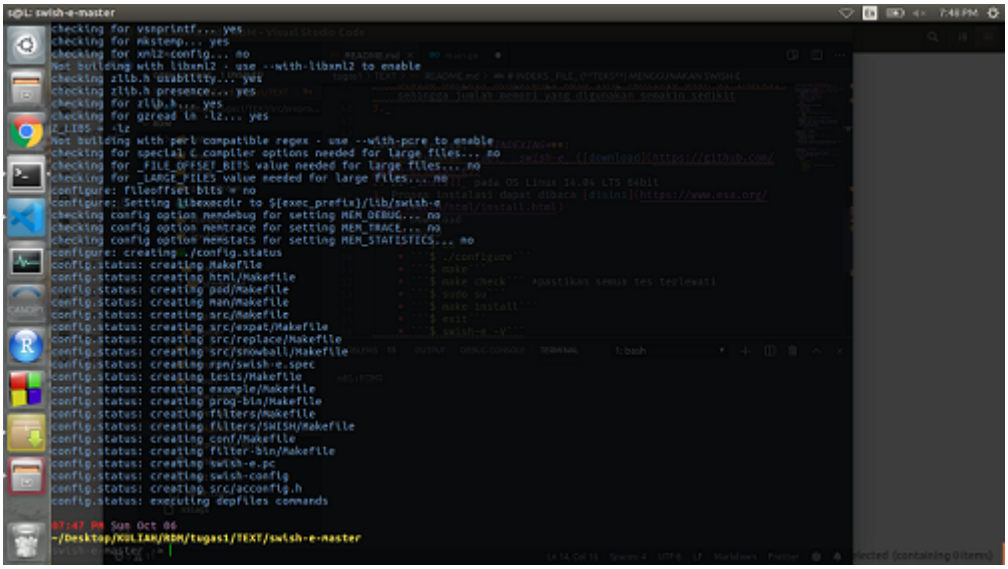
6. Cuplikan ketika proses *STEMMING* berlangsung:



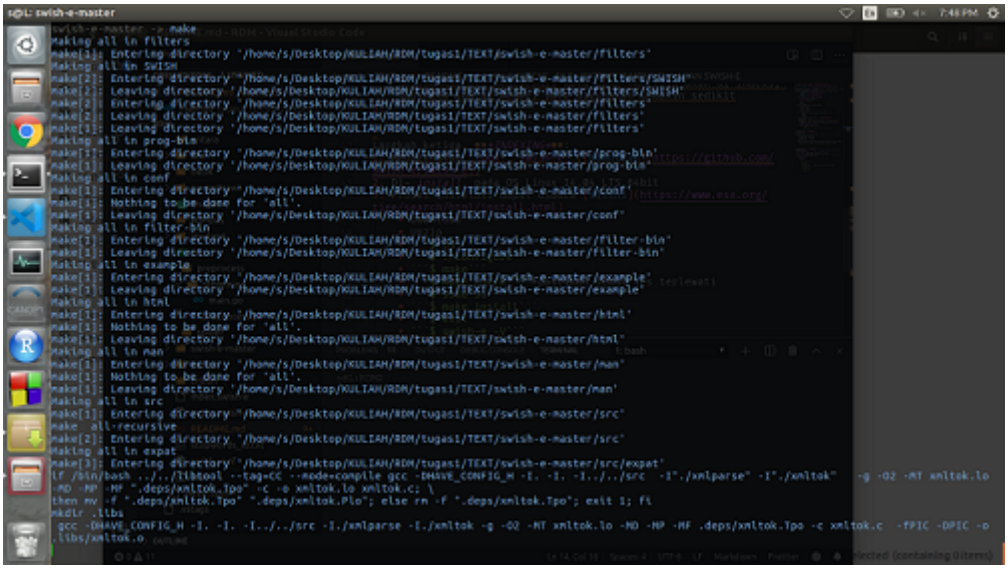
7. Hasil *STEMMING* ditaruh di folder:
    - `ALL_2/kompas`
    - `ALL_2/tempo`
    - `ALL_2/antara`

Langkah ketiga, ***INDEXING***:

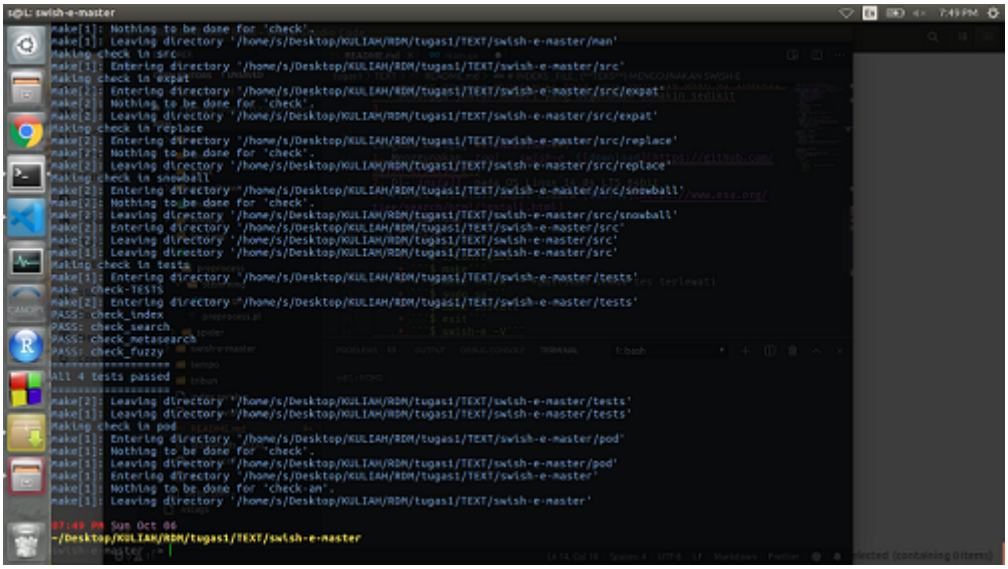
1. Menggunakan *tool swish-e* (download)
2. Di-*install* pada OS Linux 16.04 LTS 64bit
3. Proses instalasi dapat dibaca disini
    - Download
    - Unzip
    - `$ cd`
    - `$ ./configure`



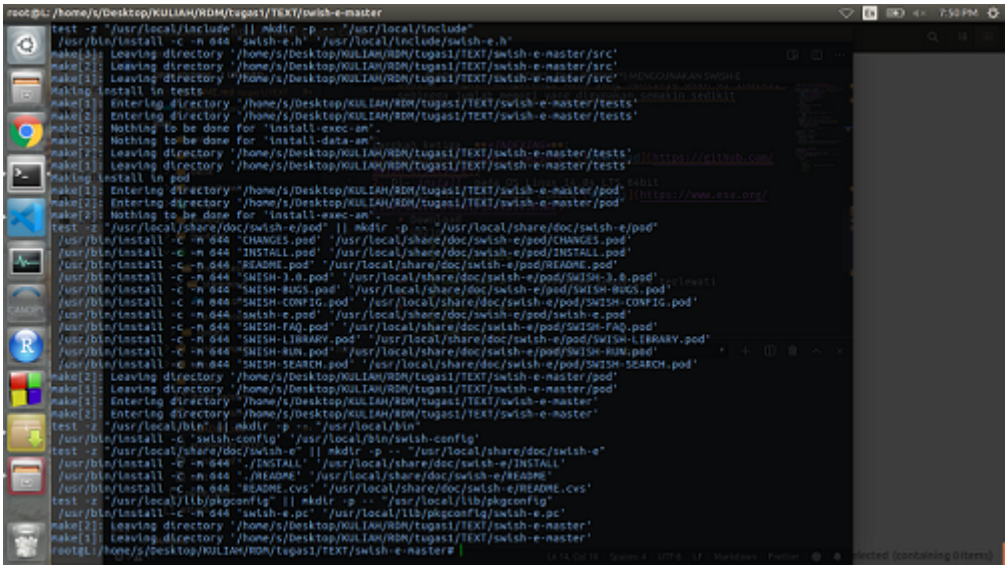
- `$ make`



- `$ make check` *pastikan semua tes berhasil

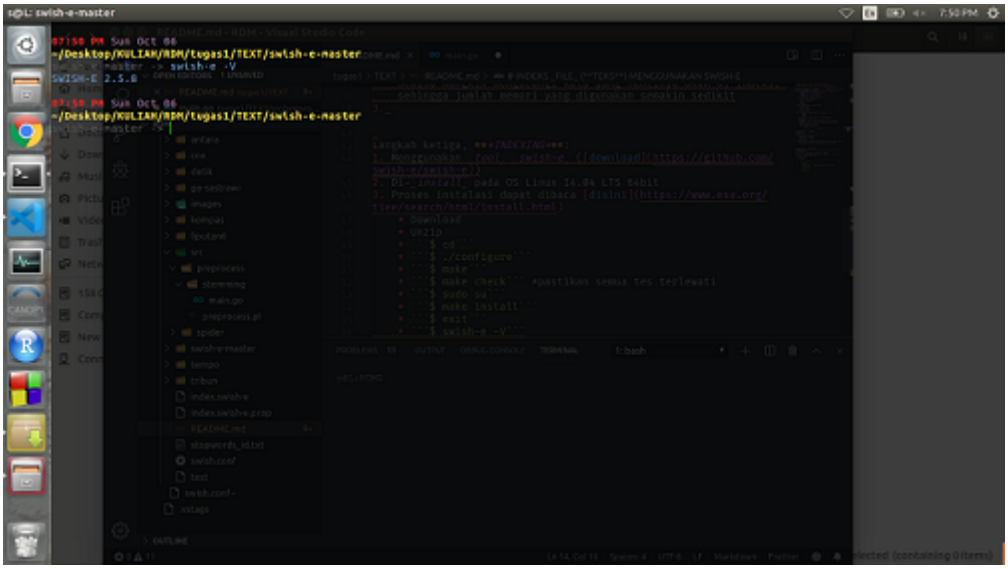


- `$ sudo su`
- `$ make install`



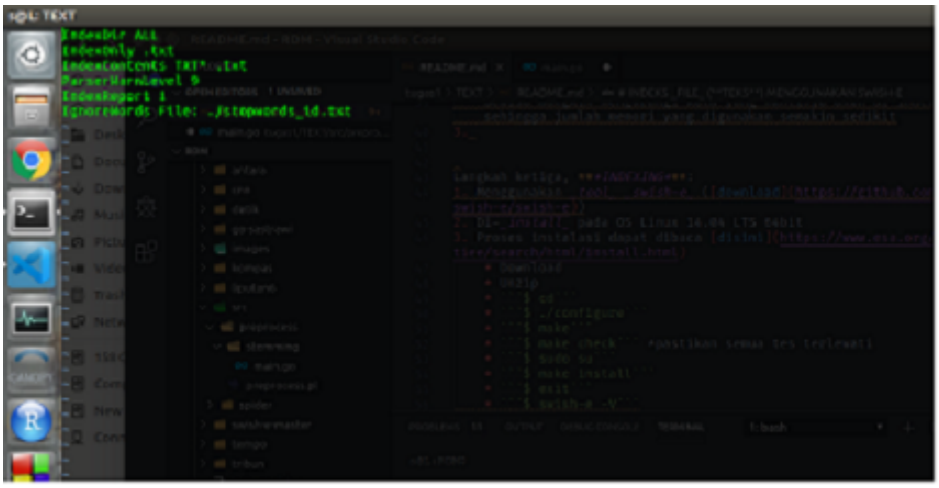
- `$ exit`

- `$ swish-e -V` SWISH-E 2.5.8

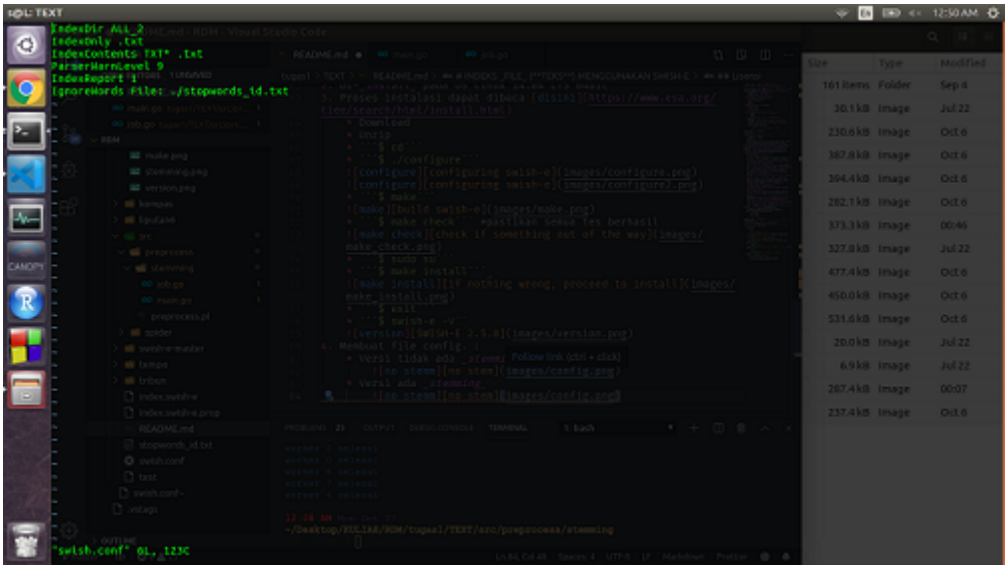


4. Membuat file config. : **BACA LEBIH LANJUT**

   - Versi tidak ada *stemming*



   - Versi ada *stemming*



5. Konfigurasi proses peng-indeks-an tercantum dalam *file* config yang telah dibuat: ***swish.conf***. *Perlu diperhatikan adanya pemangkasan kata umum (*common words*)

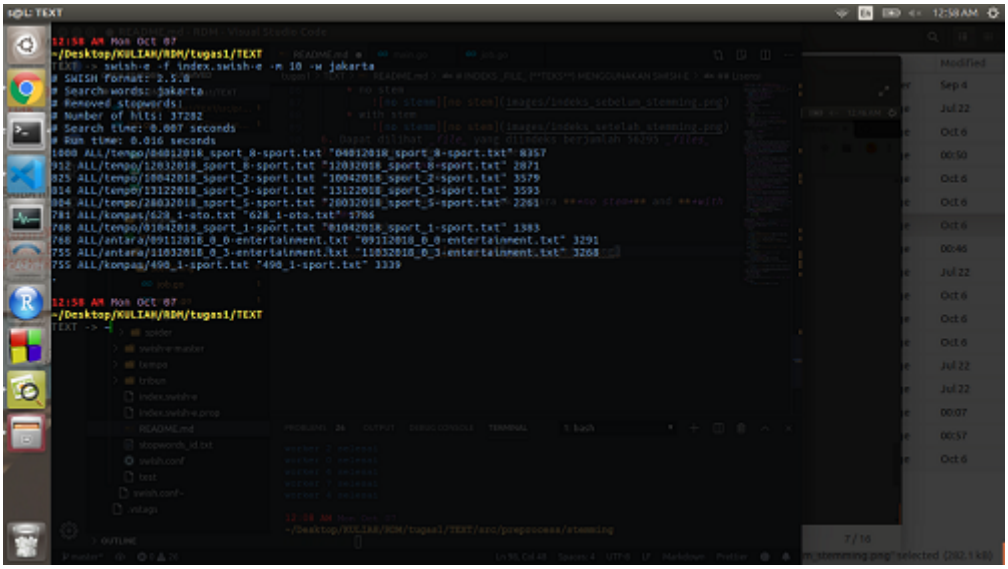
(*stopwords*) menggunakan data *stopwords* bahasa indonesia (**stopwords_id.txt**) yang dapat di-*download* melalui link ini. Data *Stopwords* ini didapat pada *website* sites.google.com/site/kevinbouge/stopwords-lists.

```
IndexFile ./indeks.swish-e
IndexDir ALL_2
IndexOnly .txt
IndexContents TXT* .txt
ParserWarnLevel 9
IndexReport 1
IgnoreWords File: ./stopwords_id.txt
```

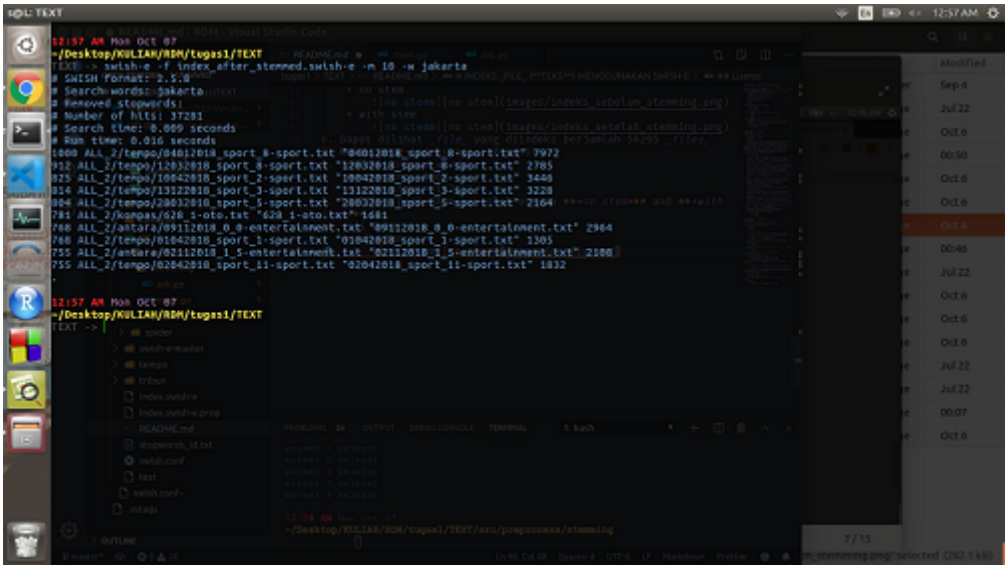
swish.conf

- no stem
  `IndexDir ALL`
- with stem
  `IndexDir ALL_2`

6. Dapat dilihat *file* yang diindeks berjumlah **56295** *files*
7. Waktu peng-indeks-an: **~10 menit**
8. Jumlah *Unique words*:
   - no stem: **~186K**
   - with stem: **~162K**
9. Terlihat perbedaan mencolok antara ***no stem*** dan ***with stem*** dalam hal jumlah *Unique words*
10. Uji Coba *search* sebuah kata:
    - no stem



- with stem

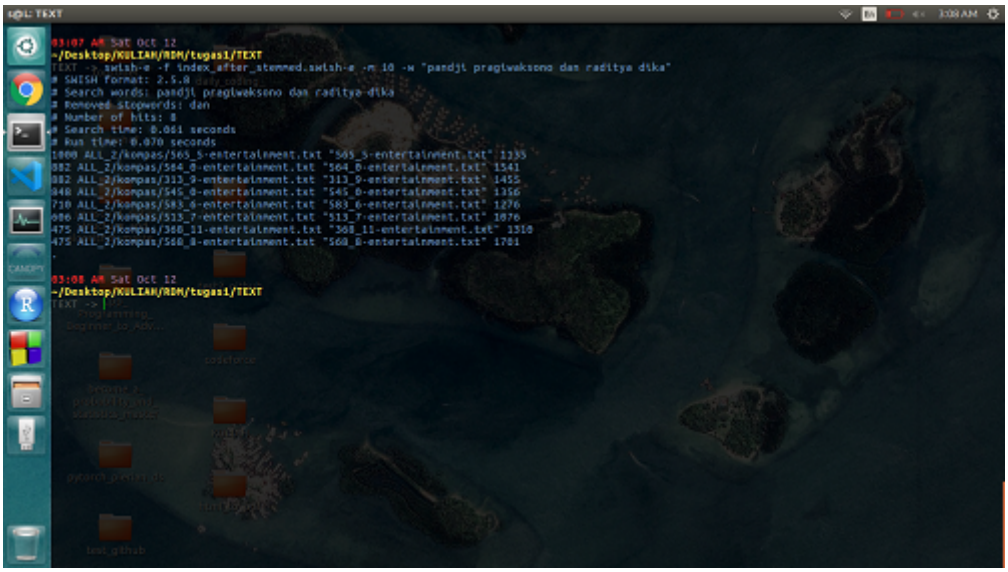


- Perbedaanya tidak terlalu jauh dari hal jumlah dokumen yang berhasil ditemukan menggunakan kata tersebut.

11. Jika ingin mencari dokumen dengan jumlah kata lebih dari satu, maka gunakan tanda petik dua ("[query]"), contohnya `swish-e -f [index.swish-e] -m 10 -w "jakarta bandung surabaya"`. *Query* tersebut akan mencari dokumen yang mengandung kata **jakarta AND bandung AND surabaya**. Jika dalam *query* tersebut ada kata yang terdapat dalam daftar *stopword*, maka akan dihapus/dilewatkan/tidak di proses. Argument dari parameter `-f` adalah nama file hasil *indexing* (*default*: `index.swish-e`).

    - *Query*: "**jakarta bandung surabaya**" atau **jakarta AND bandung AND surabaya**
      Hit: 390

    - *Query*: "**pandji pragiwaksono dan raditya dika**"
      Hit: 8



Dapat di-intepret sebagai: "*cari dokumen yang mengandung kata* ***pandji AND pragiwaksono AND dan AND raditya AND dika***"
Kata '**dan**' tidak diproses karena termasuk dalam daftar *stopwords*

- *Query*: "**raditya dika AND NOT pandji pragiwaksono**" dan "**raditya dika OR pandji pragiwaksono**"
  Hit: 0 dan 71
  Dapat di-intepret sebagai:
  - "*cari dokumen yang mengandung kata* ***raditya AND dika tapi tidak ada kata pandji AND pragiwaksono***"
  - "*cari dokumen yang mengandung kata* ***raditya AND dika atau pandji AND pragiwaksono***"

Parameter `-m [n]` digunakan untuk menampilkan *top n* dokumen, misalkan `-m 10` menampilkan 10 dokumen teratas.

12. Arti dari Output (format):

    - Kolom-1: Ranking
    - Kolom-2: Lokasi dokumen/*file*
    - Kolom-3: Judul/Nama dokumen/*file*
    - Kolom-4: *Size*/Ukuran dokumen (dalam *bytes*) **BACA LEBIH LANJUT**

---Download---

Semua file yang digunakan dalam projek ini dapat di unduh pada link berikut UNDUH FILE

github

Versi enak dilihat -> link

Data

File konfigurasi dapat dilihat di sini