

## ***Sentiment Analysis 2.0***



setelah berhasil membuat program sederhana untuk melakukan sentiment analysis pada tugas sebelumnya, kamu semakin tertarik untuk mengetahui lebih dalam tentang sentiment analysis. Kamu ingin melangkah lebih jauh dan menggunakan dataset yang telah disediakan untuk melakukan sentiment analysis yang lebih kompleks. Tujuan utamanya adalah **untuk memprediksi sentimen suatu kalimat, apakah itu memiliki sentimen positif atau negatif**. Untuk mencapai tujuan tersebut, ada beberapa hal berikut yang harus diimplementasikan.

### **Implementasi:**

Terdapat 4 menu yang harus kamu kerjakan:

#### **1. prediksi orientasi sentiment kalimat**

pada menu ini, user akan diminta untuk memasukkan suatu kalimat, nantinya program akan diminta untuk mengeluarkan hasil prediksi yang dihasilkan dari function predict.

function predict bekerja dengan cara menjumlahkan skor-skor dari tiap kata pada kalimat tersebut. skor dari setiap kata dapat dilihat dari berkas

**positive.tsv** atau **negative.tsv**. Lalu, berikut adalah dua kasus yang dapat terjadi:

- Apabila kata tidak terdapat pada kedua berkas, kata tersebut **diberikan nilai 0**.
- Apabila kata terdapat pada kedua berkas, skor kata tersebut dihitung dengan **rumus  $\text{score}(w) = \text{pos\_score}(w) + \text{neg\_score}(w)$**

Hasil akhir prediksi akan ditentukan dengan skor akhir. apabila skor akhir positif, maka hasil prediksinya positif begitu juga sebaliknya. **hasil prediksi adalah positif jika  $\geq 0$  dan negatif jika  $< 0$ .**

## 2. evaluasi model prediksi

pada menu ini, program akan melakukan prediksi (instruksi poin 1) pada semua kalimat dalam file *data\_sentiment\_mobil.csv*. Implementasi fungsi eval sudah disediakan. **hasil prediksi adalah positif jika  $\geq 0$  dan negatif jika  $< 0$ .** Output dari menu ini adalah nilai akurasi model dalam bentuk desimal 2 angka di belakang koma.

## 3. word count positif

Pada menu ini, program akan menghitung jumlah kemunculan setiap kata yang tidak ada di stopwords pada file sentimen mobil. Setelah itu, program akan menampilkan daftar 20 kata positif yang paling sering muncul.

## 4. word count negatif

Pada menu ini, program akan menghitung jumlah kemunculan setiap kata yang tidak ada di stopwords pada file sentimen mobil. Setelah itu, program akan menampilkan daftar 20 kata negatif yang paling sering muncul.

## Deskripsi File:

1. **data\_sentiment\_mobil.csv:** berisi kalimat-kalimat beserta sentiment-nya dengan format *<kalimat>:<sentiment>*. Contoh:

<b>kalimat</b>	<b>sentiment</b>
Kalo menurut gw enak pajero si	pos
punya mobil altis tidak awet	neg

2. **positive.tsv**: berisi daftar kata beserta skor sentimen positif dengan format *<kata>:<skor>*.

kata	skor
silakan	1
percaya	2
makasih	4
terbaik	5

3. **negative.tsv**: berisi daftar kata beserta skor sentimen negatif dengan format *<kata>:<skor>*.

kata	skor
doang	-1
berumur	-2
pusing	-4
mahal	-5

4. **stopwords.txt**: berisi daftar stopwords, yaitu kata-kata yang umumnya dianggap tidak memiliki makna penting dalam pemrosesan teks. Contoh stopwords adalah "*dan*", "*atau*", "*di*", dan "*adalah*". Stopwords biasanya dihapus dari teks sebelum dilakukan analisis lebih lanjut karena dianggap tidak berkontribusi terhadap makna utama teks dan hanya menambah beban komputasi. Contoh:

```
stopwords_list = ['saya', 'adalah', 'dan', 'di', 'ini']
```

Kalimat Sebelum Diproses:

"**Saya** sedang belajar pemrosesan teks **dan** analisis data **di** kampus **ini**."

Kalimat Setelah Diproses:

"Sedang belajar pemrosesan teks analisis data kampus."

5. **tp3 template.py**: template kode. Kalian bisa mulai dengan membaca kode dari bagian `if __name__ == "__main__"` lalu telusuri setiap pemanggilan function untuk mendapatkan gambaran alur proses program dan input & output yang diharapkan pada setiap function. Setelah itu, kerjakan dengan mengisi function yang hanya memiliki tulisan **#TODO**

### Test Case:

Catatan:

- Kalian dibebaskan untuk membuat keyword untuk opsi dan bagaimana tampilan program dengan syarat semua opsi harus bekerja dengan baik.
- Tulisan berwarna **biru** menandakan input dari user dan **merah** menandakan output.
- program juga tidak harus looping terus menerus untuk menerima input user. (setelah selesai di satu menu boleh langsung stop)
- Kutip (") pada input dan output bukanlah hal yang wajib ada. Kutip hanya pembatas pada string input dan output agar terlihat spasi di awal dan akhir kalimat

```
.:Pilih Menu:.  
[1] prediksi orientasi sentiment kalimat  
[2] evaluasi model prediksi  
[3] word count positif  
[4] word count negatif  
Menu yang dipilih = 1  
Tuliskan kalimat review tentang mobil Anda!  
>> Saya memakai Honda Jazz GK5 tahun 2014 ( pertama meluncur ) .  
Mobil nya bagus dan enak sesuai moto nya menyenangkan untuk  
dikendarai  
prediksi = pos  
  
.:Pilih Menu:.  
[1] prediksi orientasi sentiment kalimat  
[2] evaluasi model prediksi  
[3] word count positif  
[4] word count negatif  
Menu yang dipilih = 2  
akurasi model prediksi = 60.0 persen  
  
.:Pilih Menu:.  
[1] prediksi orientasi sentiment kalimat  
[2] evaluasi model prediksi
```

```
[3] word count positif
[4] word count negatif
Menu yang dipilih = 3
kata positif diurutkan berdasarkan jumlah kemunculan dalam
pasangan (kata:jumlah)
irit:100
nyaman:67
banyak:33
cocok:31
beli:30
bakar:29
suka:29
mantap:28
desain:26
menurut:25
lumayan:25
mudah:24
oke:24
jadi:22
dalam:20
tenaga:20
sampai:18
lincah:18
pelayanan:17
satu:17
```

..Pilih Menu:.

```
[1] prediksi orientasi sentiment kalimat
[2] evaluasi model prediksi
[3] word count positif
[4] word count negatif
Menu yang dipilih = 4
```

```
kata negatif diurutkan berdasarkan jumlah kemunculan dalam
pasangan (kata:jumlah)
irit:100
tidak:50
mahal:35
banyak:33
cocok:31
bahan:31
beli:30
jauh:30
bakar:29
suka:29
cuma:27
jelek:26
cukup:25
lumayan:25
```

bandel:24  
mudah:24  
kurang:23  
jadi:22  
juragan:22  
pun:22

## Pembahasan

kalimat sebelum diproses:

“Saya memakai Honda Jazz GK5 tahun 2014 ( pertama meluncur ) .  
Mobil nya bagus dan enak sesuai moto nya menyenangkan untuk  
dikendarai”

kalimat setelah diproses:

“memakai honda jazz gk5 2014 ( pertama meluncur ) . sesuai moto  
- - - - - +3 -  
menyenangkan dikendarai”  
+1 -

total skor adalah 4 (skor  $\geq 0$ ) sehingga diklasifikasikan sebagai sentimen **pos**

keterangan:

“-” menandakan kata/karakter tidak ada pada *positive.tsv* dan *negative.tsv*

## Syarat Pengumpulan:

1. Kumpulkan sebelum waktu deadline tiba. **Keterlambatan tidak akan ditoleransi.**
2. Format penamaan file: **TP3\_[KodeAsdos]\_[NPM]\_[Nama].py**

**Plagiarisme tidak akan ditoleransi.** Anda diperbolehkan untuk berdiskusi dengan teman terkait ide implementasi tugas ini dengan menuliskan NPM dan nama kolaborator anda. Namun perlu diingat bahwa implementasi kode tetap dilakukan secara individu. Jika suatu saat terbukti melakukan kecurangan/plagiarisme akan mendapat sanksi berupa nilai 0 pada TP ini dan mungkin sanksi lain sesuai dengan penilaian dosen.

Good luck! 🐧 🦉