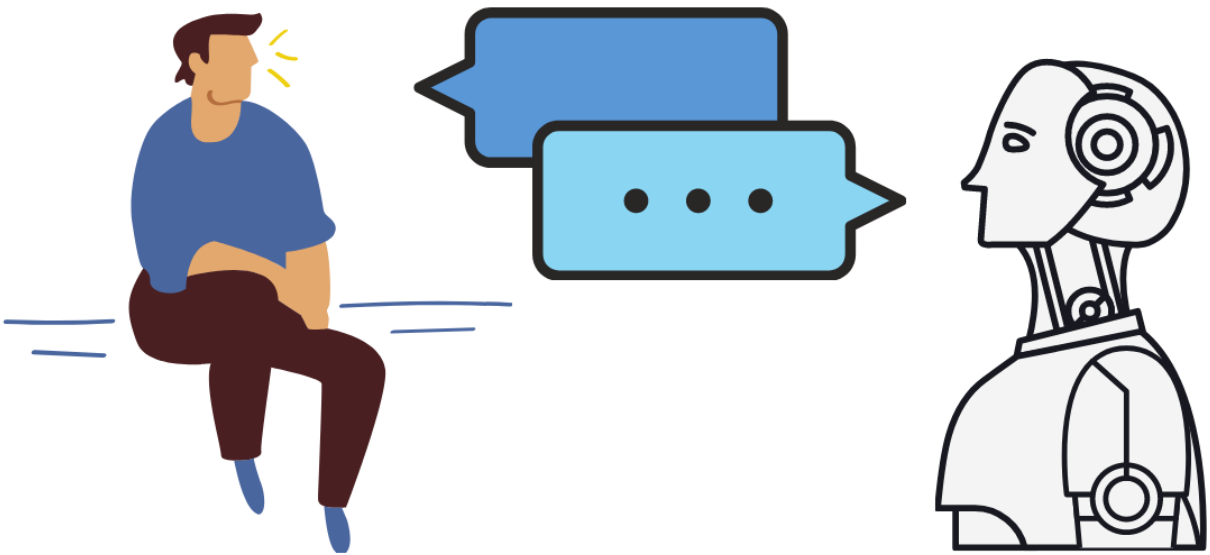
Pengolahan Bahasa Alami

Analisis Sentimen

Menggunakan *Logistic Regression*

Maria Veronica Claudia M., M.T.

Semester Genap 2020/2021



PETUNJUK:

1. Kerjakan berkelompok, masing-masing kelompok terdiri dari dua orang. Agar mudah, gunakan kelompok tugas akhir Anda.
2. Untuk kali ini, Anda diberi *template* dan artikel. Baca terlebih dahulu artikel tersebut (sekilas saja) untuk mengetahui gambaran tentang apa yang akan Anda kerjakan. Tentunya Anda tidak dapat langsung menyalin seluruh kode program yang diberikan di artikel tersebut karena *template* yang diberikan sudah dimodifikasi. Jadi, Anda tetap harus memahami bagian kode mana dari artikel tersebut yang dapat Anda pakai, hanya dapat dicontoh, dan tidak dipakai lagi.
3. *Rename* file *template.py* menjadi P09_noKelompok.py.
4. Jawab pula pertanyaan berwarna hijau, simpan jawaban Anda dalam file M09_noKelompok.txt.
5. Satukan seluruh pekerjaan Anda ke dalam folder dengan format penamaan T09_noKelompok. Kumpulkan dalam bentuk zip. Satu kelompok kumpulkan satu file saja.

Pendahuluan

Pada praktikum kali ini, Anda akan melakukan eksperimen untuk topik analisis sentimen. *Dataset* yang digunakan adalah *review* film dari IMDB. Anda telah diberi *dataset* yang sudah berupa file CSV, silakan pergunakan *dataset* tersebut (tidak perlu unduh lagi dari artikel).

Ide klasifikasi yang digunakan diambil dari artikel yang dapat Anda akses melalui tautan berikut: https://towardsdatascience.com/sentiment-analysis-with-python-part-1-5ce197074184.

Untuk mempelajari tentang *logistic regression*, Anda dapat membaca artikel pada tautan: https://towardsdatascience.com/logistic-regression-detailed-overview-46c4da4303bc

Tujuan akhir dari eksperimen ini adalah untuk mendapatkan *positive word list* dan *negative word list* dari hasil pembelajaran *data train* menggunakan *logistic regression*. *Word lists* yang dihasilkan akan dipakai di praktikum berikutnya. Jadi, usahakan untuk mengerjakan praktikum ini sampai berhasil.

Template

Anda telah diberi *template* dengan 6 variabel global (masing-masing telah diberi penjelasan berupa komentar) dan 9 fungsi (termasuk *main function*). Seperti biasa, sambil membaca modul, bacalah pula komentar-kompentar pada *template*. Tanda '##' pada komentar berarti penjelasan, sedangkan '#' berarti Anda perlu mengisi / membuat sendiri bagian tersebut.

Fungsi

Fungsi – fungsi yang disediakan sudah terurut sesuai dengan apa yang perlu Anda kerjakan untuk melakukan analisis sentiment, yaitu sebagai berikut.

1. **Membaca file**. Gunakan fungsi *readFile* untuk mengubah file CSV menjadi *pandas dataframe.*
2. **Membersihkan teks**. Setiap teks pada *dataframe* perlu dibersihkan terlebih dahulu. Untuk membersihkan satu teks, Anda dapat melengkapi fungsi *txtCleaner*. Untuk membersihkan seluruh teks dalam *dataframe*, lengkapilah fungsi *dfCleaner*.
3. **Membagi *dataset***. Untuk membuat model dan menguji kualitas model tersebut, Anda memerlukan *data train* dan *data test*. Detail langkah-langkah pembagian dataset dapat langsung dibaca pada *template* di bagian fungsi *splitDf*. Untuk mempelajari *split dataset* lebih lanjut, silakan baca tautan berikut:
   https://www.codegrepper.com/code-examples/python/split+dataframe+into+train+and+test+python
   1. Pada *template* Anda diminta untuk membagi *dataset* berdasarkan label (positif dan negatif) terlebih dahulu sebelum membagi *data train* dan *data test*. Mengapa demikian?
   2. Apa yang mungkin akan terjadi jika hal tersebut (membagi berdasarkan kelas) tidak dilakukan?

   Pada bagian akhir fungsi ini Anda diminta untuk menggabungkan kembali data berlabel positif dan negative untuk masing-masing *dataset* (*train* dan *test*). Cara menggabungkan *dataframe* dapat Anda lihat di https://tinyurl.com/joinDataframe.

4. **Vektorisasi (pemodelan)**. Pada eksperimen ini Anda diminta untuk membuat dua pemodelan, yaitu menggunakan *binary events vectorizer* (*vectCnt*) dan *TF-IDF vectorizer* (*vectTFIDF*). Nantinya kedua bentuk pemodelan ini akan dibandingkan pengaruhnya terhadap hasil klasifikasi.
   3. Apa perbedaan *binary events* dengan *word count*?
5. **Membuat model klasifikasi menggunakan *logistic regression***. Pembuatan model klasifikasi dan pengukuran kualitas (menggunakan akurasi) ada pada fungsi *classifier*. Anda dapat mempelajari *logistic regression* menggunakan *library scikit-learn* di tautan https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LinearRegression.html. *Inverse of regularization strength* yang digunakan adalah 0.5. Anda dapat membaca artikel pada tautan https://towardsdatascience.com/regularization-in-machine-learning-76441ddcf99a untuk mengetahui tentang *regularization*.

4. Pemodelan teks yang manakah yang menghasilkan model klasifikasi dengan akurasi lebih tinggi?

6. **Membangun daftar kata positif dan negatif**. Bagian ini dapat Anda lihat di artikel referensi. Karena perlu modifikasi (ada pembulatan 3 desimal), Anda tetap perlu untuk memahami apa yang dilakukan kode program dari artikel saat membuat *feature_to_coef*.
   5. Apa keluaran dari fungsi *zip()* pada *python*?
   6. Pada kode program pembuatan *feature_to_coef* terdapat bagian *word:coef*. Apa artinya?

   Di bagian akhir, Anda perlu menyimpan daftar kata yang telah tersusun berdasarkan nilai *coef*-nya. Untuk mempelajari fungsi *sorted*, Anda dapat melihat contoh-contoh penggunaannya di https://www.geeksforgeeks.org/sorted-function-python/.
   7. Apa maksud dari x[1] pada bagian *key*? Jika bingung, bandingkan hasilnya dengan menggunakan x[0].
   8. Mengapa perlu *reverse = True*?

Hasil Akhir

Dengan mengikuti langkah-langkah di atas dan komentar pada *template*, tentunya sambil membaca artikel-artikel dan *tutorial* yang diberikan, seharusnya Anda akan menghasilkan *dictionary* yang berisi daftar kata dengan sentimen positif dan negatif. Lihatlah daftar kata yang dihasilkan, lalu jawab pertanyaan berikut.

9. Dengan mengabaikan adanya kata OOV Bahasa Inggris dan *stop words*, apakah kata-kata dalam daftar tersebut masuk akal? Mengapa?
10. Jika tidak masuk akal, apa yang perlu ditambahkan untuk mengatasi hal tersebut? (Diasumsikan kata-kata OOV dan *stop words* sudah dihapus).

== Selamat Mengerjakan ==