

KETENTUAN UJIAN PERTEMUAN 5

INFO PROJECT PRAKTIKUM KECERDASAN ARTIFICIAL

1. Wajib membuat project analisis data dengan ketentuan sebagai berikut :

- Project yang dibuat berupa pembuatan model dengan memilih salah satu topik dibawah ini: A. Convolutional Neural Network B. Natural Language Processing C. Supervised / Unsupervised Learning D. Autoencoder
 - Project ini bersifat individual. TIDAK DIKERJAKAN SECARA KELOMPOK
 - Tidak boleh menggunakan project ACT / LA yang telah diberikan saat praktikum sebelumnya!!
2. Project dikumpulkan pada hari H Ujian Praktikum melalui Google Drive yang di share dengan link (Jika link tidak bisa diakses maka tidak dinilai).
- Folder pada google drive berisi : • Project Colab (bentuk .ipynb) • Manual Book (bentuk pdf) • Video Presentasi Project. Video wajib menampilkan wajah praktikan dan wajib menggunakan kemeja (Bentuk mp4, berisi penjelasan logika program serta output program dengan durasi 5-15 menit) • Data Diri (bentuk txt, berisi : Nama, Npm, Kelas, dan Kesan Pesan (terhadap PJ, asisten & LabTi))
 - Pengumpulan link di upload pada Activity Ujian, jangan lupa untuk aksesnya di ubah menjadi PUBLIC
 - Ubah link gdrive menjadi fotmat : UJIAN_NAMA LENGKAP_KELAS

Jika ada pertanyaan silakan menghubungi PJ. Terimakasih dan Selamat Ujian ✨

Library

```
!pip install gensim # Menginstall library Gensim pada python
```

```
Requirement already satisfied: gensim in /usr/local/lib/python3.10/dist-packages (4.3.2)
Requirement already satisfied: numpy>=1.18.5 in /usr/local/lib/python3.10/dist-packages (from gensim) (1.23.5)
Requirement already satisfied: scipy>=1.7.0 in /usr/local/lib/python3.10/dist-packages (from gensim) (1.11.3)
Requirement already satisfied: smart-open>=1.8.1 in /usr/local/lib/python3.10/dist-packages (from gensim) (6.4.0)
```

```
import pandas as pd #Import library pandas dan berikan alias pd
import gensim #Import library gensim
from gensim.models import word2vec #import modul word2vec dari library gensim
import time #import library time untuk mengukur waktu eksekusi program
import multiprocessing #import library multiprocessing untuk melakukan paralel processing
from datetime import timedelta #import class timedelta dari library datetime untuk melakukan operasi pada waktu
```

Load Data

df = pd.read_csv('simpsons_script_lines.csv')#membaca file CSV yaitu "simpsons_script_lines.csv" dan menyimpannya ke dalam variabel df.shape#Menampilkan dimensi dari dataframe df, yaitu jumlah baris dan kolom, menggunakan atribut shape dari dataframe

```
(158271, 2)
```

df.head()#menampilkan 5 baris pertama dari dataframe df menggunakan fungsi head dari dataframe

	raw_character_text	normalized_text
0	Miss Hoover	no actually it was a little of both sometimes ...
1	Lisa Simpson	wheres mr bergstrom
2	Miss Hoover	i dont know although id sure like to talk to h...
3	Lisa Simpson	that life is worth living
4	Edna Krabappel-Flanders	the polls will be open from now until the end ...

Data Cleaning

df.isnull().sum()#menampilkan jumlah nilai null pada setiap kolom dataframe df menggunakan fungsi isnull() dan sum() dari dataframe

```
raw_character_text    17522
normalized_text       26184
```

```
dtype: int64
```

```
df = df.dropna().reset_index(drop=True)#menghapus baris yang memiliki nilai null dari dataframe df menggunakan fungsi dropna() dari
df.isnull().sum()#Menampilkan jumlah nilai null pada setiap kolom dataframe df setelah baris yang memiliki nilai null dihapus menggu
```

```
raw_character_text    0
normalized_text       0
dtype: int64
```

▼ Membuat corpus

```
corpus_text = "\n".join(df['normalized_text'])# Menggabungkan semua teks dari kolom normalized_text pada dataframe df menjadi satu :
```

```
# Menyimpan corpus ke dalam file 'corpus.txt'
corpus_path = 'corpus.txt'#menyimpan path file untuk menyimpan corpus teks yang telah digabungkan ke dalam variabel corpus_path
with open(corpus_path, 'w', encoding='utf-8') as f:
    f.write(corpus_text)#Membuka file dengan path yang telah ditentukan sebelumnya,menuliskan corpus teks yang telah digabungkan ke
```

▼ Training Model

```
start_time = time.time()#menyimpan waktu awal training model word2vec ke variabel start_time menggunakan fungsi time.time()
print('Training Word2Vec Model...')
sentences = word2vec.LineSentence(corpus_path)#membaca corpus teks dari file yang telah disimpan sebelumnya dan mengubahnya menjadi
w2v_model = word2vec.Word2Vec(sentences, vector_size=300, workers=multiprocessing.cpu_count())#melakukan training model word2vec pada
w2v_model.save('model_word2vec_300_model')#menyimpan model word2vec yang telah dilatih kedalam file dengan nama "model_word2vec_300_
finish_time = time.time()#menyimpan waktu akhir training model word2vec kedalam variabel finish_time menggunakan fungsi time.time()
```

```
Training Word2Vec Model...
```

▼ Test

```
w2v_model.wv.similarity('woman','man')#mengukur kemiripan antara kata woman dan man
```

```
0.75294936
```

```
w2v_model.wv.most_similar('chicken')#mencari kata yang paling mirip dengan chicken
```

```
[('salad', 0.8629200458526611),
 ('ruthless', 0.8594102263450623),
 ('ems', 0.8537583351135254),
 ('doctors', 0.8520733118057251),
 ('fermented', 0.8511450290679932),
 ('role', 0.8503414392471313),
 ('sweat', 0.8497821688652039),
 ('potato', 0.8482204079627991),
 ('fuzzy', 0.8471310138702393),
 ('shaped', 0.846598744392395)]
```

```
w2v_model.wv.most_similar(positive=['homer'])#mencari kata yang paling mirip dengan homer
```

```
[('marge', 0.8359345197677612),
 ('bart', 0.8318659663200378),
 ('lisa', 0.7785618305206299),
 ('abe', 0.7086857557296753),
 ('grampa', 0.6999842524528503),
 ('milhouse', 0.6658316254615784),
 ('moe', 0.6375312805175781),
 ('mrs', 0.6371076107025146),
 ('mr', 0.6338191032409668),
 ('homie', 0.6192125082015991)]
```

```
w2v_model.wv.most_similar(positive=['marge'])#mencari kata yang paling mirip dengan marge
```

```
[('homer', 0.8359344005584717),
 ('lisa', 0.786033570766449),
 ('bart', 0.7632092237472534),
 ('homie', 0.7419790029525757),
 ('honey', 0.7177525162696838),
 ('moe', 0.700082004070282),
 ('dad', 0.6990009546279907),
 ('milhouse', 0.6910751461982727),
 ('maggie', 0.6746408939361572),
 ('son', 0.6644091010093689)]
```

```
w2v_model.wv.similarity('maggie', 'baby')#mengukur kemiripan kata maggie dan baby
```

```
0.5851635
```

```
w2v_model.wv.most_similar(positive=['woman', 'homer'], negative=['marge'], topn=3)#mencari kata yang paling mirip dengan woman dan h
```

```
[('man', 0.689825177192688),  
 ('person', 0.646925687789917),  
 ('hero', 0.613666296005249)]
```

```
w2v_model.wv.most_similar(positive=['woman','king'], negative=['homer'])#Mencari kata yang paling mirip dengan woman dan king tapi t
```

```
[('evil', 0.7194916605949402),  
 ('american', 0.7107086181640625),  
 ('star', 0.7018318176269531),  
 ('role', 0.6975564360618591),  
 ('mans', 0.695903480052948),  
 ('model', 0.6926491260528564),  
 ('famous', 0.687658965587616),  
 ('birth', 0.6840470433235168),  
 ('human', 0.6801149249076843),  
 ('concept', 0.6762233376502991)]
```

```
w2v_model.wv.doesnt_match('homer drink milk sleep angry'.split())#mencari kata yang tidak cocok dalam daftar kata homer drink milk :
```

```
'angry'
```