**TUGAS 1**

ANALISIS SENTIMEN KESEHATAN MENTAL DENGAN DATASET “SENTIMENT ANALYSIS FOR MENTAL HEALTH”

**LAPORAN**

**Oleh:**

**Tim GCD**

**IF-A PAGI**

* **211110347 - CINDY SINTIYA**
* **211110948 - GRACE HELENA HUTAGAOL**
* **211111930 - DAVID BATE'E**



****

**PROGRAM STUDI S-1**

**FAKULTAS INFORMATIKA**

**TEKNIK INFORMATIKA**

**UNIVERITAS MIKROSKIL**

**2024/2025**

# Daftar Isi

[Judul 1](#_Toc181649829)

[Tujuan 1](#_Toc181649830)

[Detail Dataset 1](#_Toc181649831)

[1. Prediksi Stress 1](#_Toc181649832)

[1.1. Sumber Dataset 1](#_Toc181649833)

[1.2. Fitur Independen (x) 1](#_Toc181649834)

[1.3. Fitur Dependen (y) 2](#_Toc181649835)

[1.4. Ukuran Dataset 2](#_Toc181649836)

[2. Analisis Sentimen Kesehatan Mental 2](#_Toc181649837)

[2.1. Sumber Dataset 2](#_Toc181649838)

[2.2. Fitur Independen (x) 2](#_Toc181649839)

[2.3. Fitur Dependen (y) 2](#_Toc181649840)

[2.4. Ukuran Dataset 2](#_Toc181649841)

[Requirements 2](#_Toc181649842)

[Preprocessing & Train-Test Split 4](#_Toc181649843)

[1. Prediksi Stress 4](#_Toc181649844)

[1.1. Data Cleaning 4](#_Toc181649845)

[1.2. Feature Encoding 5](#_Toc181649846)

[1.3. Feature Selection 5](#_Toc181649847)

[1.4. Train-Test Splitting 5](#_Toc181649848)

[2. Analisis Sentimen 5](#_Toc181649849)

[2.1. Data Cleaning 5](#_Toc181649850)

[2.2. Data balancing 5](#_Toc181649851)

[2.3. Feature Encoding 5](#_Toc181649852)

[2.4. Train-Test Splitting 5](#_Toc181649853)

[2.5. Tokenizer 5](#_Toc181649854)

[2.6. Pad Sequences 5](#_Toc181649855)

[2.7. Batch Data Loading 5](#_Toc181649856)

[Pemilihan Algoritma dan Evaluasi Model 5](#_Toc181649857)

[1. Prediksi Stress 5](#_Toc181649858)

[2. Analisis Sentimen 5](#_Toc181649859)

[Pengembangan Aplikasi 5](#_Toc181649860)

[Kesimpulan 5](#_Toc181649861)

[Link Video Presentasi 5](#_Toc181649862)

[Link Notebook 5](#_Toc181649863)

[Link GitHub/ Source Code Lengkap 5](#_Toc181649864)

# Judul

Prediksi Stress dan Analisis Sentimen Kesehatan Mental dengan Dataset “*Mental Health Dataset*” dan “*Sentiment Analysis For Mental Health*”

Penjelasan dan Alasan...

Relavansi dengan ml...

# Tujuan

1. Memprediksi apakah seseorang mengalami stress atau tidak melalui beberapa variabel, seperti pekerjaan, sosialisasi, riwayat gangguan mental sebelumnya, dan sebagainya.
2. Memprediksi seberapa besar kemungkinan seseorang mengalami gangguan mental, seperti depresi, gangguan kecemasan, kecenderungan bunuh diri, dan sebagainya.

# Detail Dataset

## Prediksi Stress

### Sumber Dataset

[kaggle.com/datasets/bhavikjikadara/mental-health-dataset](https://www.kaggle.com/datasets/bhavikjikadara/mental-health-dataset)

### Fitur Independen (x)

|  |  |  |
| --- | --- | --- |
| *Timestamp* | : | waktu pengiriman form |
| *Gender* | : | jenis kelamin (*Male/ Female*) |
| *Country* | : | negara |
| *Occupation* | : | pekerjaan (*Business/ Corporate/ Housewife/ Student/ Other*) |
| *self\_employed* | : | wiraswasta (*Yes/ No*) |
| *family\_history* | : | apakah ada anggota keluarga dengan riwayat gangguan mental (*Yes/ No*) |
| *treatment* | : | apakah sudah pernah mencari pengobatan untuk masalah gangguan kesehatan mental (*Yes/ No*) |
| *Days\_Indoors* | : | sudah berapa lama stay dirumah saja (*Go out Every day/ 1-14 days/ 15-30 days/ 31-60 days/ More than 2 months*) |
| *Changes\_Habits* | : | perubahan kebiasaan (*Yes/ No/ Maybe*) |
| *Mental\_Health\_History* | : | riwayat gangguan mental (*Yes/ No/ Maybe*) |
| *Mood\_Swings* | : | perubahan mood/ suasana hati (*Low/ Medium/ High*) |
| *Coping\_Struggles* | : | apakah berjuang/ berusaha mengatasinya (*Yes/ No*) |
| *Work\_Interest* | : | sangat terikat dengan pekerjaan (*Yes/ No/ Maybe*) |
| *Social\_Weakness* | : | lemah dalam bersosialisasi (*Yes/ No/ Maybe*) |
| *mental\_health\_interview* | : | apakah akan menceritakan terkait masalah kesehatan saat *interview* kerja (*Yes/ No/ Maybe*) |
| *care\_options* | : | kepedulian terhadap masalah (*Yes/ No/ Not Sure*) |

### Fitur Dependen (y)

*Growing\_Stress* (*Yes/ No/ Maybe*)

### Ukuran Dataset

292.364 baris data, 17 kolom fitur

## Analisis Sentimen Kesehatan Mental

### Sumber Dataset

[kaggle.com/datasets/suchintikasarkar/sentiment-analysis-for-mental-health](https://www.kaggle.com/datasets/suchintikasarkar/sentiment-analysis-for-mental-health)

### Fitur Independen (x)

*statement* : sentimen dalam bentuk teks kalimat

### Fitur Dependen (y)

*status* (*Normal/ Depression/ Suicidal/ Anxiety/ Stress/ Bipolar/ Personality disorder*)

### Ukuran Dataset

53.043 baris data, 2 kolom fitur (+ 1 kolom *index*)

# Requirements

1. Versi Notebook
2. Akses [link notebook Google Colab](link) atau bisa menggunakan *file* “ML\_Mental\_Health\_PT.ipynb” (model dibangun menggunakan pytorch) atau “ML\_Mental\_Health\_TF.ipynb” (model dibangun menggunakan tensorflow) yang ada di dalam folder “notebook”.

Disarankan untuk menggunakan *file* *Colab* agar menghemat waktu eksekusi dan *training* model menggunakan GPU yang telah disediakan oleh *Google Colab Compute Engine Backend*.

1. Siapkan API kaggle atau *file* dataset dalam format csv.

Untuk mengambil dataset dapat dilakukan pengunduhan secara manual melalui [link kaggle berikut](https://www.kaggle.com/datasets/bhavikjikadara/mental-health-dataset) ataupun menggunakan *file* “stress.csv” untuk prediksi stress dan juga [link kaggle berikut](https://www.kaggle.com/datasets/suchintikasarkar/sentiment-analysis-for-mental-health) ataupun menggunakan *file* “sentiments.csv” untuk analisis sentimen, yang sudah tersedia di folder “data”.

* Jika ingin mengunduh secara otomatis dari *notebook*, pastikan sudah menyiapkan API kaggle yang disimpan dalam *file* “kaggle.json”, lalu *upload* ke *runtime* *notebook* Colab.
* Jika menggunakan *file* yang sudah tersedia di folder “data”, pastikan sudah *upload* *file* dataset pada *runtime* *notebook* Colab dan melakukan *comment* dan *uncomment* pada *code* sesuai dengan video penjelasan yang ada.

1. Eksekusi 1 per 1 *cell* secara berurutan (opsional).

Jika ingin menggunakan model dari proses ini, pastikan sudah menjalankan semua *cell* (minimal hingga “Model Evaluation” untuk *save* keseluruhan model hasil *training* dan digunakan pada aplikasi *web* sebagai *pre-trained model*).

Step ini bersifat opsional karena *file* *pre-trained model* juga sudah tersedia pada folder “data” dan dapat langsung digunakan pada aplikasi *web*.

1. Versi Aplikasi *Web* (*end-user*)
2. Lakukan instalasi *library* yang tercatat dalam file “requirement.txt”

Jalankan

pip install -r requirement.txt

pada terminal dan tunggu hingga instalasi selesai.

Jika menggunakan Anaconda, pastikan *base* conda sudah aktif dengan menjalankan perintah berikut:

conda activate base

atau

C:/Users/<device name>/anaconda3/Scripts/activate

(sesuaikan dengan lokasi Anaconda perangkat Anda)

lalu jalankan:

conda install Flask Jinja2 flask-cors numpy pandas tensorflow torch scikit-learn tqdm deep-translator joblib

dan tunggu hingga instalasi selesai.

Atau bisa juga membaca file “README.md” yang ada untuk detail instalasi *library*.

1. Jika ingin mengganti model dengan yang baru di-*save* dari notebook, pastikan *file* model dan konfigurasi diletakkan di folder yang sama dengan lokasi *file* model awalnya (folder “data”) agar aplikasi dapat berjalan tanpa *error*.
2. Jalankan aplikasi web dengan perintah

python app.py

Jika menggunakan Anaconda, pastikan *base* conda sudah aktif sebelum menjalanakan aplikasi.

1. Tunggu hingga aplikasi berhasil *running*, kemudian akses [http://127.0.0.1:5000](http://127.0.0.1:5000/) di *browser* atau sesuaikan dengan *link* yang muncul di terminal.
2. Aplikasi web siap digunakan.
3. Versi Aplikasi *Mobile* (*end-user*)
4. Buka terminal (cmd/ powershell), lalu jalankan perintah

flutter pub get

1. Untuk dapat menggunakan versi *mobile*, pastikan aplikasi *web* sudah aktif dan berjalan karena akan digunakan sebagai API untuk aplikasi *mobile*.
2. Jalankan

flutter run

lalu pilih *device* yang akan digunakan untuk *running* aplikasi *mobile* (selain *device* Android, bisa juga menggunakan *browser* seperti Chrome ataupun Edge untuk menjalankan aplikasi).

1. Masukkan IP Address/ IP Host dari aplikasi *web* yang sudah berjalan (IP dapat dilihat pada terminal aplikasi *web*) melalui tombol *Setting* yang ada dibagian sudut kanan atas *interface*, lalu klik tombol “*Save and Try Again*”.
2. Tunggu hingga model selesai di-*fetch* dan di-*load* dari *backend* API *web* dan aplikasi *mobile* siap digunakan.

# Preprocessing & Train-Test Split

## Prediksi Stress

### Data Cleaning

### Feature Encoding

### Feature Selection

### Train-Test Splitting

## Analisis Sentimen

### Data Cleaning

### Data balancing

### Feature Encoding

### Train-Test Splitting

### Tokenizer

### Pad Sequences

### Batch Data Loading

# Pemilihan Algoritma dan Evaluasi Model

## Prediksi Stress

## Analisis Sentimen

# Pengembangan Aplikasi

# Kesimpulan

# Link Video Presentasi

[Akses video presentasi GCD - Mental Health Sentiment Analysis disini](link)

# Link Notebook

[Akses notebook GCD - Mental Health Sentiment Analysis disini](link)

# Link GitHub/ Source Code Lengkap

[Akses GitHub GCD - Mental Health Sentiment Analysis disini](link)