```
# Step 1: Installing Libraries
!pip install -q sentence-transformers
!pip install -q nltk
!pip install -q scikit-learn
!pip install -q tensorflow
!pip install -q keras==2.11.0
!pip install tensorflow==2.11.0
%pip install --upgrade numpy
%pip install shap
%pip install optuna
```

```
huggingface/tokenizers: The current process just got forked, after
parallelism has already been used. Disabling parallelism to avoid
deadlocks...
To disable this warning, you can either:
	- Avoid using `tokenizers` before the fork if possible
	- Explicitly set the environment variable
TOKENIZERS_PARALLELISM=(true | false)
huggingface/tokenizers: The current process just got forked, after
parallelism has already been used. Disabling parallelism to avoid
deadlocks...
To disable this warning, you can either:
	- Avoid using `tokenizers` before the fork if possible
	- Explicitly set the environment variable
TOKENIZERS_PARALLELISM=(true | false)
huggingface/tokenizers: The current process just got forked, after
parallelism has already been used. Disabling parallelism to avoid
deadlocks...
To disable this warning, you can either:
	- Avoid using `tokenizers` before the fork if possible
	- Explicitly set the environment variable
TOKENIZERS_PARALLELISM=(true | false)
huggingface/tokenizers: The current process just got forked, after
parallelism has already been used. Disabling parallelism to avoid
deadlocks...
To disable this warning, you can either:
	- Avoid using `tokenizers` before the fork if possible
	- Explicitly set the environment variable
TOKENIZERS_PARALLELISM=(true | false)

ERROR: pip's dependency resolver does not currently take into account
all the packages that are installed. This behaviour is the source of
the following dependency conflicts.
tensorflow-macos 2.16.2 requires tensorflow==2.16.2; platform_system
== "Darwin" and platform_machine == "arm64", but you have tensorflow
2.19.0 which is incompatible.
```

```
huggingface/tokenizers: The current process just got forked, after
parallelism has already been used. Disabling parallelism to avoid
deadlocks...
To disable this warning, you can either:
	- Avoid using `tokenizers` before the fork if possible
	- Explicitly set the environment variable
TOKENIZERS_PARALLELISM=(true | false)

ERROR: pip's dependency resolver does not currently take into account
all the packages that are installed. This behaviour is the source of
the following dependency conflicts.
tensorflow 2.19.0 requires keras>=3.5.0, but you have keras 2.11.0
which is incompatible.
tensorflow-macos 2.16.2 requires tensorflow==2.16.2; platform_system
== "Darwin" and platform_machine == "arm64", but you have tensorflow
2.19.0 which is incompatible.

huggingface/tokenizers: The current process just got forked, after
parallelism has already been used. Disabling parallelism to avoid
deadlocks...
To disable this warning, you can either:
	- Avoid using `tokenizers` before the fork if possible
	- Explicitly set the environment variable
TOKENIZERS_PARALLELISM=(true | false)

ERROR: Could not find a version that satisfies the requirement
tensorflow==2.11.0 (from versions: 2.16.0rc0, 2.16.1, 2.16.2,
2.17.0rc0, 2.17.0rc1, 2.17.0, 2.17.1, 2.18.0rc0, 2.18.0rc1, 2.18.0rc2,
2.18.0, 2.18.1, 2.19.0rc0, 2.19.0, 2.20.0rc0)
ERROR: No matching distribution found for tensorflow==2.11.0

huggingface/tokenizers: The current process just got forked, after
parallelism has already been used. Disabling parallelism to avoid
deadlocks...
To disable this warning, you can either:
	- Avoid using `tokenizers` before the fork if possible
	- Explicitly set the environment variable
TOKENIZERS_PARALLELISM=(true | false)

Requirement already satisfied: numpy in
/Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/
site-packages (1.26.4)
Collecting numpy
  Using cached numpy-2.2.6-cp310-cp310-macosx_14_0_arm64.whl.metadata
(62 kB)
Using cached numpy-2.2.6-cp310-cp310-macosx_14_0_arm64.whl (5.3 MB)
Installing collected packages: numpy
  Attempting uninstall: numpy
    Found existing installation: numpy 1.26.4
    Uninstalling numpy-1.26.4:
      Successfully uninstalled numpy-1.26.4
ERROR: pip's dependency resolver does not currently take into account
all the packages that are installed. This behaviour is the source of
the following dependency conflicts.
gradio 3.45.2 requires numpy~=1.0, but you have numpy 2.2.6 which is
incompatible.
gradio 3.45.2 requires websockets<12.0,>=10.0, but you have websockets
15.0.1 which is incompatible.
scipy 1.10.1 requires numpy<1.27.0,>=1.19.5, but you have numpy 2.2.6
which is incompatible.
tensorflow 2.19.0 requires keras>=3.5.0, but you have keras 2.12.0
which is incompatible.
tensorflow 2.19.0 requires numpy<2.2.0,>=1.26.0, but you have numpy
2.2.6 which is incompatible.
tensorflow 2.19.0 requires tensorboard~=2.19.0, but you have
tensorboard 2.12.3 which is incompatible.
tensorflow-macos 2.12.0 requires numpy<1.24,>=1.22, but you have numpy
2.2.6 which is incompatible.
Successfully installed numpy-2.2.6
Note: you may need to restart the kernel to use updated packages.

huggingface/tokenizers: The current process just got forked, after
parallelism has already been used. Disabling parallelism to avoid
deadlocks...
To disable this warning, you can either:
     - Avoid using `tokenizers` before the fork if possible
     - Explicitly set the environment variable
TOKENIZERS_PARALLELISM=(true | false)

Requirement already satisfied: shap in
/Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/
site-packages (0.48.0)
Requirement already satisfied: numpy in
/Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/
site-packages (from shap) (2.2.6)
Requirement already satisfied: scipy in
/Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/
site-packages (from shap) (1.10.1)
Requirement already satisfied: scikit-learn in
/Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/
site-packages (from shap) (1.6.1)
Requirement already satisfied: pandas in
/Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/
site-packages (from shap) (2.0.1)
Requirement already satisfied: tqdm>=4.27.0 in
/Users/anmoldevansh/Library/Python/3.10/lib/python/site-packages (from
shap) (4.65.0)
Requirement already satisfied: packaging>20.9 in
/Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/
site-packages (from shap) (21.3)
```

```
Requirement already satisfied: slicer==0.0.8 in
/Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/
site-packages (from shap) (0.0.8)
Requirement already satisfied: numba>=0.54 in
/Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/
site-packages (from shap) (0.61.2)
Requirement already satisfied: cloudpickle in
/Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/
site-packages (from shap) (3.1.1)
Requirement already satisfied: typing-extensions in
/Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/
site-packages (from shap) (4.14.1)
Requirement already satisfied: llvmlite<0.45,>=0.44.0dev0 in
/Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/
site-packages (from numba>=0.54->shap) (0.44.0)
Requirement already satisfied: pyparsing!=3.0.5,>=2.0.2 in
/Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/
site-packages (from packaging>20.9->shap) (3.0.9)
Requirement already satisfied: python-dateutil>=2.8.2 in
/Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/
site-packages (from pandas->shap) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in
/Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/
site-packages (from pandas->shap) (2023.3)
Requirement already satisfied: tzdata>=2022.1 in
/Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/
site-packages (from pandas->shap) (2023.3)
Requirement already satisfied: six>=1.5 in
/Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/
site-packages (from python-dateutil>=2.8.2->pandas->shap) (1.16.0)
Requirement already satisfied: joblib>=1.2.0 in
/Users/anmoldevansh/Library/Python/3.10/lib/python/site-packages (from
scikit-learn->shap) (1.2.0)
Requirement already satisfied: threadpoolctl>=3.1.0 in
/Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/
site-packages (from scikit-learn->shap) (3.2.0)
Collecting numpy (from shap)
  Using cached numpy-1.26.4-cp310-cp310-macosx_11_0_arm64.whl.metadata
(61 kB)
Using cached numpy-1.26.4-cp310-cp310-macosx_11_0_arm64.whl (14.0 MB)
Installing collected packages: numpy
  Attempting uninstall: numpy
    Found existing installation: numpy 2.2.6
    Uninstalling numpy-2.2.6:
      Successfully uninstalled numpy-2.2.6
ERROR: pip's dependency resolver does not currently take into account
all the packages that are installed. This behaviour is the source of
the following dependency conflicts.
gradio 3.45.2 requires websockets<12.0,>=10.0, but you have websockets
```

```
15.0.1 which is incompatible.
tensorflow 2.19.0 requires keras>=3.5.0, but you have keras 2.12.0
which is incompatible.
tensorflow 2.19.0 requires tensorboard~=2.19.0, but you have
tensorboard 2.12.3 which is incompatible.
tensorflow-macos 2.12.0 requires numpy<1.24,>=1.22, but you have numpy
1.26.4 which is incompatible.
Successfully installed numpy-1.26.4
Note: you may need to restart the kernel to use updated packages.

huggingface/tokenizers: The current process just got forked, after
parallelism has already been used. Disabling parallelism to avoid
deadlocks...
To disable this warning, you can either:
      - Avoid using `tokenizers` before the fork if possible
      - Explicitly set the environment variable
TOKENIZERS_PARALLELISM=(true | false)

Requirement already satisfied: optuna in
/Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/
site-packages (4.4.0)
Requirement already satisfied: alembic>=1.5.0 in
/Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/
site-packages (from optuna) (1.16.4)
Requirement already satisfied: colorlog in
/Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/
site-packages (from optuna) (6.9.0)
Requirement already satisfied: numpy in
/Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/
site-packages (from optuna) (1.26.4)
Requirement already satisfied: packaging>=20.0 in
/Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/
site-packages (from optuna) (21.3)
Requirement already satisfied: sqlalchemy>=1.4.2 in
/Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/
site-packages (from optuna) (2.0.42)
Requirement already satisfied: tqdm in
/Users/anmoldevansh/Library/Python/3.10/lib/python/site-packages (from
optuna) (4.65.0)
Requirement already satisfied: PyYAML in
/Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/
site-packages (from optuna) (6.0.1)
Requirement already satisfied: Mako in
/Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/
site-packages (from alembic>=1.5.0->optuna) (1.3.10)
Requirement already satisfied: typing-extensions>=4.12 in
/Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/
site-packages (from alembic>=1.5.0->optuna) (4.14.1)
Requirement already satisfied: tomli in
/Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/
```

```
site-packages (from alembic>=1.5.0->optuna) (2.2.1)
Requirement already satisfied: pyparsing!=3.0.5,>=2.0.2 in
/Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/
site-packages (from packaging>=20.0->optuna) (3.0.9)
Requirement already satisfied: MarkupSafe>=0.9.2 in
/Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/
site-packages (from Mako->alembic>=1.5.0->optuna) (2.1.2)
Note: you may need to restart the kernel to use updated packages.
```

```python
# Step 2: Import Libraries
import pandas as pd
import numpy as np
import re
import string
import nltk
import matplotlib.pyplot as plt
import seaborn as sns
import tensorflow as tf

# If running in Google Colab, uncomment the next line:
# import google.colab.files as files


from sentence_transformers import SentenceTransformer
from sklearn.model_selection import train_test_split
from sklearn.metrics import classification_report, accuracy_score,
confusion_matrix
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Dropout, BatchNormalization
from tensorflow.keras.callbacks import EarlyStopping, CSVLogger,
ReduceLROnPlateau
from tensorflow.keras.regularizers import l2

nltk.download('punkt')
```

```
[nltk_data] Downloading package punkt to
[nltk_data]     /Users/anmoldevansh/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
```

```
True
```

```python
# Step 3: Load & Preprocess Dataset
# Upload your CSV file manually or mount Google Drive if needed

#from google.colab import files
#uploaded = files.upload()

df = pd.read_csv("depression_dataset_reddit.csv")

# Renaming columns for simplicity
```

```
df.columns = ['text', 'label']
df.head()
```

```
                                                text  label
0  we understand that most people who reply immed...      1
1  welcome to r depression s check in post a plac...      1
2  anyone else instead of sleeping more when depr...      1
3  i ve kind of stuffed around a lot in my life d...      1
4  sleep is my greatest and most comforting escap...      1
```

```
import string

def clean_text(text):
    text = text.lower()
    text = re.sub(r"http\S+|www\S+|https\S+", '', text, 
flags=re.MULTILINE)
    text = re.sub(r'\@w+|\#','', text)
    text = re.sub(r'[^\x00-\x7f]',r'', text)  # Remove non-ASCII
    text = re.sub(r'[%s]' % re.escape(string.punctuation), '', text)
    text = re.sub(r'\s+', ' ', text).strip()
    return text

df['clean_text'] = df['text'].apply(clean_text)
df = df[df['clean_text'].str.strip() != ""]
df.head()
```

```
                                                text  label  \
0  we understand that most people who reply immed...      1   
1  welcome to r depression s check in post a plac...      1   
2  anyone else instead of sleeping more when depr...      1   
3  i ve kind of stuffed around a lot in my life d...      1   
4  sleep is my greatest and most comforting escap...      1   

                                          clean_text  
0  we understand that most people who reply immed...  
1  welcome to r depression s check in post a plac...  
2  anyone else instead of sleeping more when depr...  
3  i ve kind of stuffed around a lot in my life d...  
4  sleep is my greatest and most comforting escap...  
```

```
sbert_model = SentenceTransformer('all-MiniLM-L6-v2')
embeddings = sbert_model.encode(df['clean_text'].tolist(), 
show_progress_bar=True)
X = np.array(embeddings)
y = df['label'].values
```

{"model_id":"af87111f401340559a93c9d16047972a","version_major":2,"vers
ion_minor":0}

```
/Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/
site-packages/torch/nn/modules/module.py:1762: FutureWarning:
```

```
`encoder_attention_mask` is deprecated and will be removed in version
4.55.0 for `BertSdpaSelfAttention.forward`.
  return forward_call(*args, **kwargs)
```

```python
from sklearn.model_selection import train_test_split

# Step 1: Split into Train (80%) and Temp (20%)
X_train, X_temp, y_train, y_temp = train_test_split(
    X, y, test_size=0.2, random_state=42, stratify=y
)

# Step 2: Split Temp into Validation (10%) and Test (10%)
X_val, X_test, y_val, y_test = train_test_split(
    X_temp, y_temp, test_size=0.5, random_state=42, stratify=y_temp
)

# Confirm the shapes
print("Train:", X_train.shape)
print("Validation:", X_val.shape)
print("Test:", X_test.shape)
```

```
Train: (6184, 384)
Validation: (773, 384)
Test: (774, 384)
```

```python
# Step 7: Train Model with CSV Logging, ReduceLR, EarlyStopping

model = Sequential([
    Dense(384, activation='relu', kernel_regularizer=l2(0.001),
input_shape=(X.shape[1],)),
    BatchNormalization(),
    Dropout(0.3),
    Dense(256, activation='relu', kernel_regularizer=l2(0.001)),
    BatchNormalization(),
    Dropout(0.3),
    Dense(128, activation='relu', kernel_regularizer=l2(0.001)),
    BatchNormalization(),
    Dropout(0.2),
    Dense(64, activation='relu', kernel_regularizer=l2(0.001)),
    Dropout(0.2),
    Dense(1, activation='sigmoid')
])

model.compile(optimizer=tf.keras.optimizers.Adam(learning_rate=2e-4),
loss='binary_crossentropy', metrics=['accuracy'])
model.summary()
```

```
WARNING:absl:At this time, the v2.11+ optimizer
`tf.keras.optimizers.Adam` runs slowly on M1/M2 Macs, please use the
legacy Keras optimizer instead, located at
`tf.keras.optimizers.legacy.Adam`.
```

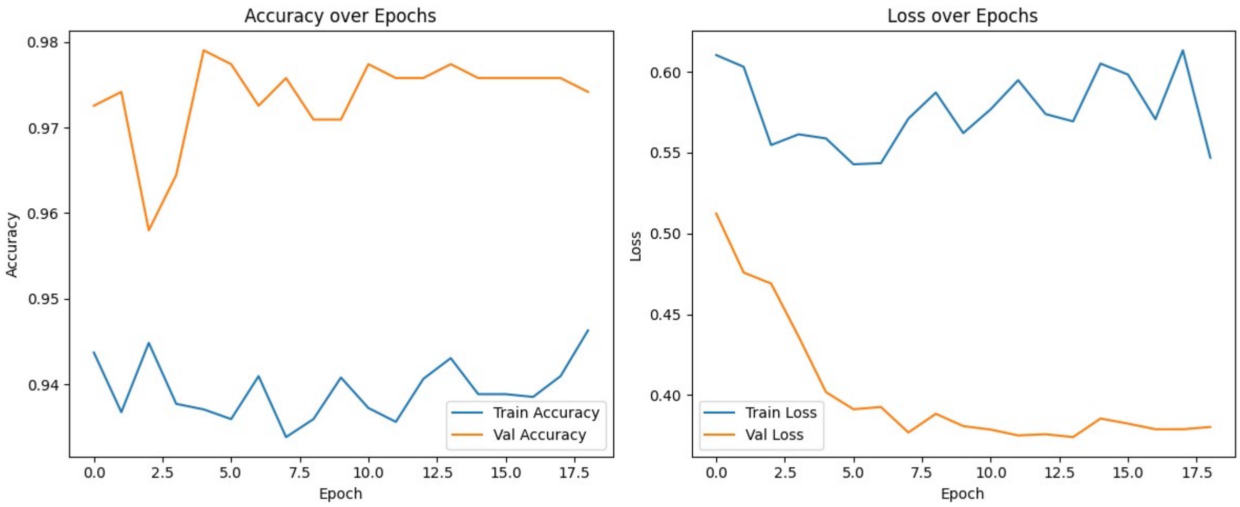
```
WARNING:absl:There is a known slowdown when using v2.11+ Keras
optimizers on M1/M2 Macs. Falling back to the legacy Keras optimizer,
i.e., `tf.keras.optimizers.legacy.Adam`.

Model: "sequential_1"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_5 (Dense)             (None, 384)               147840

 batch_normalization_3 (Batc  (None, 384)              1536
 hNormalization)

 dropout_4 (Dropout)         (None, 384)               0

 dense_6 (Dense)             (None, 256)               98560

 batch_normalization_4 (Batc  (None, 256)              1024
 hNormalization)

 dropout_5 (Dropout)         (None, 256)               0

 dense_7 (Dense)             (None, 128)               32896

 batch_normalization_5 (Batc  (None, 128)              512
 hNormalization)

 dropout_6 (Dropout)         (None, 128)               0

 dense_8 (Dense)             (None, 64)                8256

 dropout_7 (Dropout)         (None, 64)                0

 dense_9 (Dense)             (None, 1)                 65

=================================================================
Total params: 290,689
Trainable params: 289,153
Non-trainable params: 1,536
_________________________________________________________________
```

```
# Step 7: Training Model with CSV Logging
from tensorflow.keras.callbacks import EarlyStopping,
ReduceLROnPlateau, CSVLogger

# Callbacks
csv_logger = CSVLogger("training_log.csv", append=False)
early_stop = EarlyStopping(monitor="val_loss", patience=5,
restore_best_weights=True)
reduce_lr = ReduceLROnPlateau(monitor="val_loss", patience=3,
factor=0.5, verbose=1)

# Model Training
history = model.fit(
    X_train, y_train,
    validation_data=(X_val, y_val),
    epochs=50,
    batch_size=32,
    callbacks=[csv_logger, early_stop, reduce_lr],
    verbose=1
)
```

```
Epoch 1/50
194/194 [==============================] - 8s 41ms/step - loss: 0.6104
- accuracy: 0.9437 - val_loss: 0.5123 - val_accuracy: 0.9725 - lr:
2.0000e-04
Epoch 2/50
194/194 [==============================] - 6s 31ms/step - loss: 0.6031
- accuracy: 0.9368 - val_loss: 0.4758 - val_accuracy: 0.9742 - lr:
2.0000e-04
Epoch 3/50
194/194 [==============================] - 6s 31ms/step - loss: 0.5547
- accuracy: 0.9449 - val_loss: 0.4690 - val_accuracy: 0.9580 - lr:
2.0000e-04
Epoch 4/50
194/194 [==============================] - 6s 30ms/step - loss: 0.5613
- accuracy: 0.9377 - val_loss: 0.4363 - val_accuracy: 0.9645 - lr:
2.0000e-04
Epoch 5/50
194/194 [==============================] - 6s 30ms/step - loss: 0.5588
- accuracy: 0.9371 - val_loss: 0.4018 - val_accuracy: 0.9790 - lr:
2.0000e-04
Epoch 6/50
194/194 [==============================] - 6s 30ms/step - loss: 0.5428
- accuracy: 0.9360 - val_loss: 0.3913 - val_accuracy: 0.9774 - lr:
2.0000e-04
Epoch 7/50
194/194 [==============================] - 6s 30ms/step - loss: 0.5435
- accuracy: 0.9410 - val_loss: 0.3927 - val_accuracy: 0.9725 - lr:
2.0000e-04
Epoch 8/50
194/194 [==============================] - 6s 30ms/step - loss: 0.5710
- accuracy: 0.9339 - val_loss: 0.3769 - val_accuracy: 0.9758 - lr:
2.0000e-04
Epoch 9/50
194/194 [==============================] - 6s 30ms/step - loss: 0.5872
- accuracy: 0.9360 - val_loss: 0.3885 - val_accuracy: 0.9709 - lr:
2.0000e-04
```

```
Epoch 10/50
194/194 [==============================] - 6s 30ms/step - loss: 0.5621
- accuracy: 0.9408 - val_loss: 0.3808 - val_accuracy: 0.9709 - lr:
2.0000e-04
Epoch 11/50
194/194 [==============================] - ETA: 0s - loss: 0.5768 -
accuracy: 0.9373
Epoch 11: ReduceLROnPlateau reducing learning rate to
9.999999747378752e-05.
194/194 [==============================] - 6s 30ms/step - loss: 0.5768
- accuracy: 0.9373 - val_loss: 0.3787 - val_accuracy: 0.9774 - lr:
2.0000e-04
Epoch 12/50
194/194 [==============================] - 6s 31ms/step - loss: 0.5948
- accuracy: 0.9356 - val_loss: 0.3751 - val_accuracy: 0.9758 - lr:
1.0000e-04
Epoch 13/50
194/194 [==============================] - 6s 30ms/step - loss: 0.5739
- accuracy: 0.9407 - val_loss: 0.3758 - val_accuracy: 0.9758 - lr:
1.0000e-04
Epoch 14/50
194/194 [==============================] - 6s 31ms/step - loss: 0.5693
- accuracy: 0.9431 - val_loss: 0.3740 - val_accuracy: 0.9774 - lr:
1.0000e-04
Epoch 15/50
194/194 [==============================] - 6s 30ms/step - loss: 0.6051
- accuracy: 0.9389 - val_loss: 0.3855 - val_accuracy: 0.9758 - lr:
1.0000e-04
Epoch 16/50
194/194 [==============================] - 6s 31ms/step - loss: 0.5983
- accuracy: 0.9389 - val_loss: 0.3824 - val_accuracy: 0.9758 - lr:
1.0000e-04
Epoch 17/50
194/194 [==============================] - ETA: 0s - loss: 0.5707 -
accuracy: 0.9386
Epoch 17: ReduceLROnPlateau reducing learning rate to
4.999999873689376e-05.
194/194 [==============================] - 6s 30ms/step - loss: 0.5707
- accuracy: 0.9386 - val_loss: 0.3789 - val_accuracy: 0.9758 - lr:
1.0000e-04
Epoch 18/50
194/194 [==============================] - 6s 31ms/step - loss: 0.6133
- accuracy: 0.9410 - val_loss: 0.3789 - val_accuracy: 0.9758 - lr:
5.0000e-05
Epoch 19/50
194/194 [==============================] - 6s 31ms/step - loss: 0.5469
- accuracy: 0.9463 - val_loss: 0.3803 - val_accuracy: 0.9742 - lr:
5.0000e-05
```

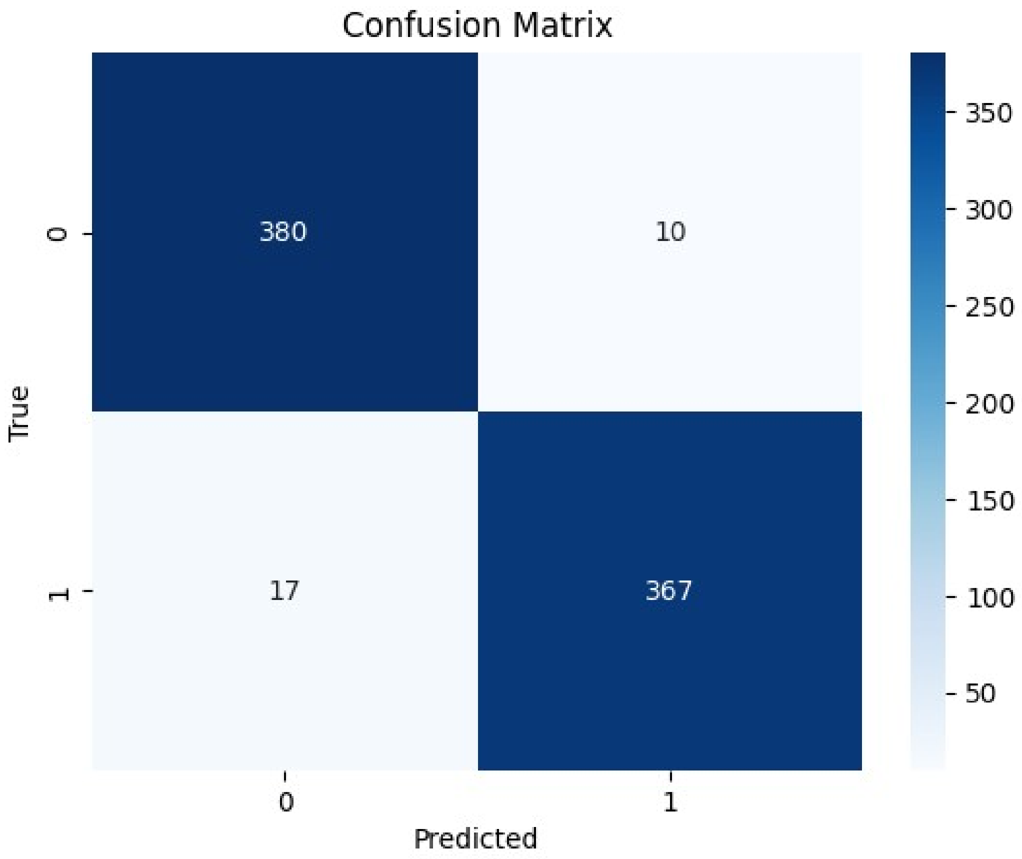
```python
# Step 8: Save Model
model.save("depression_sbert_model.keras")
pd.DataFrame(df).to_csv("depression_training_data.csv", index=False)

# Step 9: Plot Accuracy & Loss
log = pd.read_csv("training_log.csv")
plt.figure(figsize=(12,5))

plt.subplot(1,2,1)
plt.plot(log['accuracy'], label='Train Accuracy')
plt.plot(log['val_accuracy'], label='Val Accuracy')
plt.title("Accuracy over Epochs")
plt.xlabel("Epoch")
plt.ylabel("Accuracy")
plt.legend()

plt.subplot(1,2,2)
plt.plot(log['loss'], label='Train Loss')
plt.plot(log['val_loss'], label='Val Loss')
plt.title("Loss over Epochs")
plt.xlabel("Epoch")
plt.ylabel("Loss")
plt.legend()

plt.tight_layout()
plt.show()
```



```python
# Step 10: Evaluate Model
y_pred_probs = model.predict(X_test)
y_pred = (y_pred_probs > 0.5).astype(int)

print("Classification Report:\n", classification_report(y_test,
y_pred))
print("Accuracy:", accuracy_score(y_test, y_pred))
```

```
cm = confusion_matrix(y_test, y_pred)
sns.heatmap(cm, annot=True, fmt='d', cmap='Blues')
plt.title("Confusion Matrix")
plt.xlabel("Predicted")
plt.ylabel("True")
plt.show()
```

```
25/25 [==============================] - 0s 5ms/step
Classification Report:
              precision    recall  f1-score   support

           0       0.96      0.97      0.97       390
           1       0.97      0.96      0.96       384

    accuracy                           0.97       774
   macro avg       0.97      0.97      0.97       774
weighted avg       0.97      0.97      0.97       774

Accuracy: 0.9651162790697675
```



```
import optuna
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Dropout, BatchNormalization
from tensorflow.keras.callbacks import EarlyStopping
from tensorflow.keras.regularizers import l2
from tensorflow.keras.optimizers import Adam
from sklearn.metrics import accuracy_score

def create_model(trial):
    model = Sequential()

    # Input Layer
    model.add(Dense(
        trial.suggest_int('units_input', 128, 512, step=64),
        activation='relu',
        kernel_regularizer=l2(trial.suggest_float('l2_input', 1e-5,
1e-2, log=True)),
        input_shape=(X_train.shape[1],)
    ))
    model.add(BatchNormalization())
    model.add(Dropout(trial.suggest_float('dropout_input', 0.2, 0.5)))

    # Hidden Layers
    for i in range(trial.suggest_int('n_layers', 1, 3)):
        units = trial.suggest_int(f'units_l{i}', 64, 256, step=64)
        model.add(Dense(units, activation='relu',
kernel_regularizer=l2(1e-3)))
        model.add(BatchNormalization())
        model.add(Dropout(trial.suggest_float(f'dropout_l{i}', 0.2,
0.5)))

    # Output Layer
    model.add(Dense(1, activation='sigmoid'))

    model.compile(
        optimizer=Adam(learning_rate=trial.suggest_float('lr', 1e-5,
1e-3, log=True)),
        loss='binary_crossentropy',
        metrics=['accuracy']
    )
    return model

def objective(trial):
    model = create_model(trial)
    early_stop = EarlyStopping(monitor='val_loss', patience=3,
restore_best_weights=True)

    model.fit(
        X_train, y_train,
        validation_data=(X_val, y_val),
        epochs=25,
        batch_size=32,
        callbacks=[early_stop],
        verbose=0
    )

    # Evaluate on validation set
    y_val_pred = (model.predict(X_val) > 0.5).astype(int)
    return accuracy_score(y_val, y_val_pred)

# Run Optuna
study = optuna.create_study(direction='maximize')
study.optimize(objective, n_trials=20)

# Print best results
print("Best Hyperparameters:")
print(study.best_params)
print(f"Best Validation Accuracy: {study.best_value:.4f}")
```

```
[I 2025-07-30 10:27:08,934] A new study created in memory with name:
no-name-be975b49-2fea-428a-b7d6-e805eaa160ea
WARNING:absl:At this time, the v2.11+ optimizer
`tf.keras.optimizers.Adam` runs slowly on M1/M2 Macs, please use the
legacy Keras optimizer instead, located at
`tf.keras.optimizers.legacy.Adam`.
WARNING:absl:There is a known slowdown when using v2.11+ Keras
optimizers on M1/M2 Macs. Falling back to the legacy Keras optimizer,
i.e., `tf.keras.optimizers.legacy.Adam`.

25/25 [==============================] - 0s 3ms/step

[I 2025-07-30 10:28:36,823] Trial 0 finished with value:
0.9547218628719275 and parameters: {'units_input': 448, 'l2_input':
0.0007083252993547784, 'dropout_input': 0.3436145956971896,
'n_layers': 1, 'units_l0': 192, 'dropout_l0': 0.4007350285806054,
'lr': 2.4134747171092852e-05}. Best is trial 0 with value:
0.9547218628719275.
WARNING:absl:At this time, the v2.11+ optimizer
`tf.keras.optimizers.Adam` runs slowly on M1/M2 Macs, please use the
legacy Keras optimizer instead, located at
`tf.keras.optimizers.legacy.Adam`.
WARNING:absl:There is a known slowdown when using v2.11+ Keras
optimizers on M1/M2 Macs. Falling back to the legacy Keras optimizer,
i.e., `tf.keras.optimizers.legacy.Adam`.

25/25 [==============================] - 0s 4ms/step

[I 2025-07-30 10:30:35,587] Trial 1 finished with value:
0.9560155239327296 and parameters: {'units_input': 192, 'l2_input':
1.5670099151895293e-05, 'dropout_input': 0.2817652337101285,
'n_layers': 2, 'units_l0': 128, 'dropout_l0': 0.3685361025621632,
'units_l1': 64, 'dropout_l1': 0.4661791494398324, 'lr':
4.954719004237469e-05}. Best is trial 1 with value:
```

```
0.9560155239327296.
WARNING:absl:At this time, the v2.11+ optimizer
`tf.keras.optimizers.Adam` runs slowly on M1/M2 Macs, please use the
legacy Keras optimizer instead, located at
`tf.keras.optimizers.legacy.Adam`.
WARNING:absl:There is a known slowdown when using v2.11+ Keras
optimizers on M1/M2 Macs. Falling back to the legacy Keras optimizer,
i.e., `tf.keras.optimizers.legacy.Adam`.

25/25 [==============================] - 0s 4ms/step

[I 2025-07-30 10:32:04,909] Trial 2 finished with value:
0.9598965071151359 and parameters: {'units_input': 320, 'l2_input':
1.3032782267879042e-05, 'dropout_input': 0.23473734742186297,
'n_layers': 1, 'units_l0': 128, 'dropout_l0': 0.2663391983959774,
'lr': 0.00012240913202726193}. Best is trial 2 with value:
0.9598965071151359.
WARNING:absl:At this time, the v2.11+ optimizer
`tf.keras.optimizers.Adam` runs slowly on M1/M2 Macs, please use the
legacy Keras optimizer instead, located at
`tf.keras.optimizers.legacy.Adam`.
WARNING:absl:There is a known slowdown when using v2.11+ Keras
optimizers on M1/M2 Macs. Falling back to the legacy Keras optimizer,
i.e., `tf.keras.optimizers.legacy.Adam`.

25/25 [==============================] - 0s 4ms/step

[I 2025-07-30 10:34:08,976] Trial 3 finished with value:
0.9547218628719275 and parameters: {'units_input': 512, 'l2_input':
1.6738188390625357e-05, 'dropout_input': 0.37421900820014453,
'n_layers': 2, 'units_l0': 256, 'dropout_l0': 0.31654993059821546,
'units_l1': 192, 'dropout_l1': 0.4626303825758132, 'lr':
1.262924943324174e-05}. Best is trial 2 with value:
0.9598965071151359.
WARNING:absl:At this time, the v2.11+ optimizer
`tf.keras.optimizers.Adam` runs slowly on M1/M2 Macs, please use the
legacy Keras optimizer instead, located at
`tf.keras.optimizers.legacy.Adam`.
WARNING:absl:There is a known slowdown when using v2.11+ Keras
optimizers on M1/M2 Macs. Falling back to the legacy Keras optimizer,
i.e., `tf.keras.optimizers.legacy.Adam`.

25/25 [==============================] - 0s 4ms/step

[I 2025-07-30 10:35:24,059] Trial 4 finished with value:
0.9560155239327296 and parameters: {'units_input': 192, 'l2_input':
0.0011648025341158684, 'dropout_input': 0.2014260364388854,
'n_layers': 2, 'units_l0': 192, 'dropout_l0': 0.4244570802924771,
'units_l1': 192, 'dropout_l1': 0.2934936996821369, 'lr':
0.0007882513343052852}. Best is trial 2 with value:
0.9598965071151359.
```

```
WARNING:absl:At this time, the v2.11+ optimizer
`tf.keras.optimizers.Adam` runs slowly on M1/M2 Macs, please use the
legacy Keras optimizer instead, located at
`tf.keras.optimizers.legacy.Adam`.
WARNING:absl:There is a known slowdown when using v2.11+ Keras
optimizers on M1/M2 Macs. Falling back to the legacy Keras optimizer,
i.e., `tf.keras.optimizers.legacy.Adam`.

25/25 [==============================] - 0s 5ms/step

[I 2025-07-30 10:37:28,102] Trial 5 finished with value:
0.9611901681759379 and parameters: {'units_input': 384, 'l2_input':
0.006194736095885513, 'dropout_input': 0.26452030205607857,
'n_layers': 2, 'units_l0': 256, 'dropout_l0': 0.4494388363004185,
'units_l1': 256, 'dropout_l1': 0.4001718190050235, 'lr':
2.7584073718576273e-05}. Best is trial 5 with value:
0.9611901681759379.
WARNING:absl:At this time, the v2.11+ optimizer
`tf.keras.optimizers.Adam` runs slowly on M1/M2 Macs, please use the
legacy Keras optimizer instead, located at
`tf.keras.optimizers.legacy.Adam`.
WARNING:absl:There is a known slowdown when using v2.11+ Keras
optimizers on M1/M2 Macs. Falling back to the legacy Keras optimizer,
i.e., `tf.keras.optimizers.legacy.Adam`.

25/25 [==============================] - 0s 5ms/step

[I 2025-07-30 10:40:07,967] Trial 6 finished with value:
0.9495472186287193 and parameters: {'units_input': 448, 'l2_input':
0.00034337763877942065, 'dropout_input': 0.3850475271375159,
'n_layers': 3, 'units_l0': 128, 'dropout_l0': 0.3838184913250582,
'units_l1': 256, 'dropout_l1': 0.3799654200872874, 'units_l2': 128,
'dropout_l2': 0.23264311471140525, 'lr': 1.5641737488119735e-05}. Best
is trial 5 with value: 0.9611901681759379.
WARNING:absl:At this time, the v2.11+ optimizer
`tf.keras.optimizers.Adam` runs slowly on M1/M2 Macs, please use the
legacy Keras optimizer instead, located at
`tf.keras.optimizers.legacy.Adam`.
WARNING:absl:There is a known slowdown when using v2.11+ Keras
optimizers on M1/M2 Macs. Falling back to the legacy Keras optimizer,
i.e., `tf.keras.optimizers.legacy.Adam`.

25/25 [==============================] - 0s 4ms/step

[I 2025-07-30 10:41:37,822] Trial 7 finished with value:
0.9573091849935317 and parameters: {'units_input': 128, 'l2_input':
0.0003640147501407975, 'dropout_input': 0.36626024822315106,
'n_layers': 1, 'units_l0': 128, 'dropout_l0': 0.2719140352877381,
'lr': 6.405891248304641e-05}. Best is trial 5 with value:
0.9611901681759379.
WARNING:absl:At this time, the v2.11+ optimizer
```

```
`tf.keras.optimizers.Adam` runs slowly on M1/M2 Macs, please use the
legacy Keras optimizer instead, located at
`tf.keras.optimizers.legacy.Adam`.
WARNING:absl:There is a known slowdown when using v2.11+ Keras
optimizers on M1/M2 Macs. Falling back to the legacy Keras optimizer,
i.e., `tf.keras.optimizers.legacy.Adam`.

25/25 [==============================] - 0s 4ms/step

[I 2025-07-30 10:42:54,103] Trial 8 finished with value:
0.9560155239327296 and parameters: {'units_input': 512, 'l2_input':
0.0014419220102218904, 'dropout_input': 0.31750703584596995,
'n_layers': 2, 'units_l0': 64, 'dropout_l0': 0.39285671485329315,
'units_l1': 256, 'dropout_l1': 0.26779400918216745, 'lr':
0.0005907176786602695}. Best is trial 5 with value:
0.9611901681759379.
WARNING:absl:At this time, the v2.11+ optimizer
`tf.keras.optimizers.Adam` runs slowly on M1/M2 Macs, please use the
legacy Keras optimizer instead, located at
`tf.keras.optimizers.legacy.Adam`.
WARNING:absl:There is a known slowdown when using v2.11+ Keras
optimizers on M1/M2 Macs. Falling back to the legacy Keras optimizer,
i.e., `tf.keras.optimizers.legacy.Adam`.

25/25 [==============================] - 0s 5ms/step

[I 2025-07-30 10:45:32,509] Trial 9 finished with value:
0.9547218628719275 and parameters: {'units_input': 128, 'l2_input':
0.00040571440699961516, 'dropout_input': 0.3690372634827809,
'n_layers': 3, 'units_l0': 256, 'dropout_l0': 0.37107269533825504,
'units_l1': 256, 'dropout_l1': 0.3381745260682041, 'units_l2': 256,
'dropout_l2': 0.3879782458538342, 'lr': 5.693290852879642e-05}. Best
is trial 5 with value: 0.9611901681759379.
WARNING:absl:At this time, the v2.11+ optimizer
`tf.keras.optimizers.Adam` runs slowly on M1/M2 Macs, please use the
legacy Keras optimizer instead, located at
`tf.keras.optimizers.legacy.Adam`.
WARNING:absl:There is a known slowdown when using v2.11+ Keras
optimizers on M1/M2 Macs. Falling back to the legacy Keras optimizer,
i.e., `tf.keras.optimizers.legacy.Adam`.

25/25 [==============================] - 0s 6ms/step

[I 2025-07-30 10:48:11,367] Trial 10 finished with value:
0.9534282018111255 and parameters: {'units_input': 320, 'l2_input':
0.009825760593336117, 'dropout_input': 0.4807071782002792, 'n_layers':
3, 'units_l0': 256, 'dropout_l0': 0.4962530671749499, 'units_l1': 64,
'dropout_l1': 0.20608422526470557, 'units_l2': 64, 'dropout_l2':
0.4896138033398002, 'lr': 0.0002665478378028124}. Best is trial 5 with
value: 0.9611901681759379.
WARNING:absl:At this time, the v2.11+ optimizer
```

```
`tf.keras.optimizers.Adam` runs slowly on M1/M2 Macs, please use the
legacy Keras optimizer instead, located at
`tf.keras.optimizers.legacy.Adam`.
WARNING:absl:There is a known slowdown when using v2.11+ Keras
optimizers on M1/M2 Macs. Falling back to the legacy Keras optimizer,
i.e., `tf.keras.optimizers.legacy.Adam`.

25/25 [==============================] - 0s 4ms/step

[I 2025-07-30 10:49:45,723] Trial 11 finished with value:
0.9611901681759379 and parameters: {'units_input': 320, 'l2_input':
8.668293830160008e-05, 'dropout_input': 0.22052196373337862,
'n_layers': 1, 'units_l0': 64, 'dropout_l0': 0.21372970195503865,
'lr': 0.00018700861572825154}. Best is trial 5 with value:
0.9611901681759379.
WARNING:absl:At this time, the v2.11+ optimizer
`tf.keras.optimizers.Adam` runs slowly on M1/M2 Macs, please use the
legacy Keras optimizer instead, located at
`tf.keras.optimizers.legacy.Adam`.
WARNING:absl:There is a known slowdown when using v2.11+ Keras
optimizers on M1/M2 Macs. Falling back to the legacy Keras optimizer,
i.e., `tf.keras.optimizers.legacy.Adam`.

25/25 [==============================] - 0s 4ms/step

[I 2025-07-30 10:51:17,782] Trial 12 finished with value:
0.9560155239327296 and parameters: {'units_input': 384, 'l2_input':
6.273514229987279e-05, 'dropout_input': 0.2603614231648471,
'n_layers': 1, 'units_l0': 64, 'dropout_l0': 0.20181442775871206,
'lr': 0.00016995436679372036}. Best is trial 5 with value:
0.9611901681759379.
WARNING:absl:At this time, the v2.11+ optimizer
`tf.keras.optimizers.Adam` runs slowly on M1/M2 Macs, please use the
legacy Keras optimizer instead, located at
`tf.keras.optimizers.legacy.Adam`.
WARNING:absl:There is a known slowdown when using v2.11+ Keras
optimizers on M1/M2 Macs. Falling back to the legacy Keras optimizer,
i.e., `tf.keras.optimizers.legacy.Adam`.

25/25 [==============================] - 0s 4ms/step

[I 2025-07-30 10:52:51,258] Trial 13 finished with value:
0.953428201811255 and parameters: {'units_input': 256, 'l2_input':
9.148521482115951e-05, 'dropout_input': 0.2080365538201819,
'n_layers': 1, 'units_l0': 192, 'dropout_l0': 0.4783632023210368,
'lr': 0.00029695277360386886}. Best is trial 5 with value:
0.9611901681759379.
WARNING:absl:At this time, the v2.11+ optimizer
`tf.keras.optimizers.Adam` runs slowly on M1/M2 Macs, please use the
legacy Keras optimizer instead, located at
`tf.keras.optimizers.legacy.Adam`.
```

```
WARNING:absl:There is a known slowdown when using v2.11+ Keras
optimizers on M1/M2 Macs. Falling back to the legacy Keras optimizer,
i.e., `tf.keras.optimizers.legacy.Adam`.

25/25 [==============================] - 0s 5ms/step

[I 2025-07-30 10:55:04,954] Trial 14 finished with value:
0.9598965071151359 and parameters: {'units_input': 384, 'l2_input':
0.006473038331381165, 'dropout_input': 0.27565669233789797,
'n_layers': 2, 'units_l0': 64, 'dropout_l0': 0.4477316595196868,
'units_l1': 128, 'dropout_l1': 0.39308243699019807, 'lr':
3.292044122878483e-05}. Best is trial 5 with value:
0.9611901681759379.
WARNING:absl:At this time, the v2.11+ optimizer
`tf.keras.optimizers.Adam` runs slowly on M1/M2 Macs, please use the
legacy Keras optimizer instead, located at
`tf.keras.optimizers.legacy.Adam`.
WARNING:absl:There is a known slowdown when using v2.11+ Keras
optimizers on M1/M2 Macs. Falling back to the legacy Keras optimizer,
i.e., `tf.keras.optimizers.legacy.Adam`.

25/25 [==============================] - 0s 5ms/step

[I 2025-07-30 10:56:43,237] Trial 15 finished with value:
0.963777490297542 and parameters: {'units_input': 384, 'l2_input':
0.00012089189281794095, 'dropout_input': 0.30814643472714676,
'n_layers': 1, 'units_l0': 192, 'dropout_l0': 0.20640202068681404,
'lr': 9.873159956295103e-05}. Best is trial 15 with value:
0.963777490297542.
WARNING:absl:At this time, the v2.11+ optimizer
`tf.keras.optimizers.Adam` runs slowly on M1/M2 Macs, please use the
legacy Keras optimizer instead, located at
`tf.keras.optimizers.legacy.Adam`.
WARNING:absl:There is a known slowdown when using v2.11+ Keras
optimizers on M1/M2 Macs. Falling back to the legacy Keras optimizer,
i.e., `tf.keras.optimizers.legacy.Adam`.

25/25 [==============================] - 0s 7ms/step

[I 2025-07-30 10:59:35,794] Trial 16 finished with value:
0.9573091849935317 and parameters: {'units_input': 384, 'l2_input':
0.004146032158270269, 'dropout_input': 0.43562811792762574,
'n_layers': 3, 'units_l0': 256, 'dropout_l0': 0.32226456155540056,
'units_l1': 128, 'dropout_l1': 0.42910152156589776, 'units_l2': 256,
'dropout_l2': 0.21340593723091209, 'lr': 8.01440997457469e-05}. Best
is trial 15 with value: 0.963777490297542.
WARNING:absl:At this time, the v2.11+ optimizer
`tf.keras.optimizers.Adam` runs slowly on M1/M2 Macs, please use the
legacy Keras optimizer instead, located at
`tf.keras.optimizers.legacy.Adam`.
WARNING:absl:There is a known slowdown when using v2.11+ Keras
```

```
optimizers on M1/M2 Macs. Falling back to the legacy Keras optimizer,
i.e., `tf.keras.optimizers.legacy.Adam`.

25/25 [==============================] - 0s 5ms/step

[I 2025-07-30 11:01:51,828] Trial 17 finished with value:
0.9573091849935317 and parameters: {'units_input': 448, 'l2_input':
0.002803940796823519, 'dropout_input': 0.3156714104497754, 'n_layers':
2, 'units_l0': 192, 'dropout_l0': 0.2581774929376727, 'units_l1': 192,
'dropout_l1': 0.491810786965417, 'lr': 2.670969624046863e-05}. Best is
trial 15 with value: 0.963777490297542.
WARNING:absl:At this time, the v2.11+ optimizer
`tf.keras.optimizers.Adam` runs slowly on M1/M2 Macs, please use the
legacy Keras optimizer instead, located at
`tf.keras.optimizers.legacy.Adam`.
WARNING:absl:There is a known slowdown when using v2.11+ Keras
optimizers on M1/M2 Macs. Falling back to the legacy Keras optimizer,
i.e., `tf.keras.optimizers.legacy.Adam`.

25/25 [==============================] - 0s 4ms/step

[I 2025-07-30 11:04:07,346] Trial 18 finished with value:
0.9521345407503234 and parameters: {'units_input': 256, 'l2_input':
3.7234401173666664e-05, 'dropout_input': 0.3174078256719019,
'n_layers': 2, 'units_l0': 256, 'dropout_l0': 0.328495404190038 1,
'units_l1': 256, 'dropout_l1': 0.3226069296122303, 'lr':
1.008356456031962e-05}. Best is trial 15 with value:
0.963777490297542.
WARNING:absl:At this time, the v2.11+ optimizer
`tf.keras.optimizers.Adam` runs slowly on M1/M2 Macs, please use the
legacy Keras optimizer instead, located at
`tf.keras.optimizers.legacy.Adam`.
WARNING:absl:There is a known slowdown when using v2.11+ Keras
optimizers on M1/M2 Macs. Falling back to the legacy Keras optimizer,
i.e., `tf.keras.optimizers.legacy.Adam`.

25/25 [==============================] - 0s 4ms/step

[I 2025-07-30 11:05:44,882] Trial 19 finished with value:
0.9611901681759379 and parameters: {'units_input': 384, 'l2_input':
0.0001580553601275497, 'dropout_input': 0.2490302623282042 4,
'n_layers': 1, 'units_l0': 192, 'dropout_l0': 0.449630685633266 2,
'lr': 4.285435805773056e-05}. Best is trial 15 with value:
0.963777490297542.

Best Hyperparameters:
{'units_input': 384, 'l2_input': 0.00012089189281794095,
'dropout_input': 0.30814643472714676, 'n_layers': 1, 'units_l0': 192,
'dropout_l0': 0.20640202068681404, 'lr': 9.873159956295103e-05}
Best Validation Accuracy: 0.9638
```

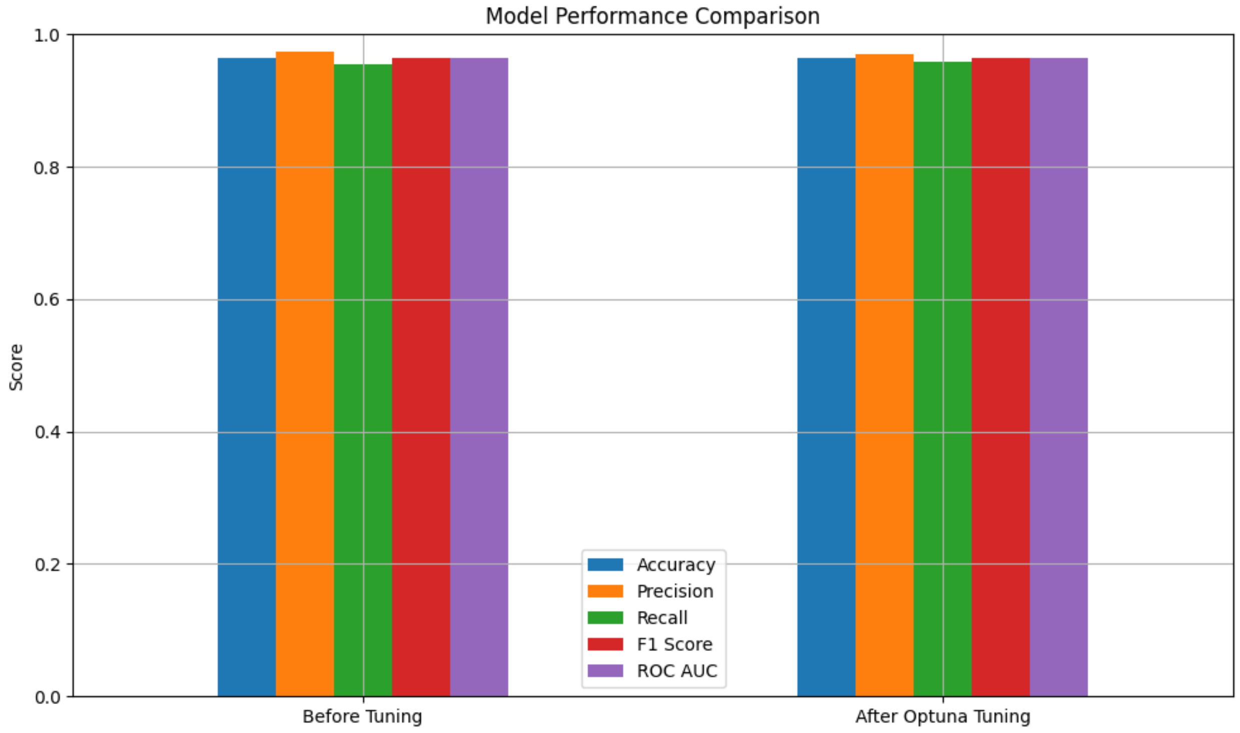
```python
# Save Optuna best model
best_trial = study.best_trial
model = Sequential()

# Input Layer
model.add(Dense(
    best_trial.params['units_input'],
    activation='relu',
    kernel_regularizer=l2(best_trial.params['l2_input']),
    input_shape=(X_train.shape[1],)
))
model.add(BatchNormalization())
model.add(Dropout(best_trial.params['dropout_input']))

# Hidden Layers
for i in range(best_trial.params['n_layers']):
    model.add(Dense(
        best_trial.params[f'units_l{i}'],
        activation='relu',
        kernel_regularizer=l2(1e-3)
    ))
    model.add(BatchNormalization())
    model.add(Dropout(best_trial.params[f'dropout_l{i}']))

# Output Layer
model.add(Dense(1, activation='sigmoid'))

model.compile(
    optimizer=Adam(learning_rate=best_trial.params['lr']),
    loss='binary_crossentropy',
    metrics=['accuracy']
)

# Fit on full training data
early_stop = EarlyStopping(monitor='val_loss', patience=3,
restore_best_weights=True)
model.fit(X_train, y_train, validation_data=(X_val, y_val), epochs=25,
batch_size=32, callbacks=[early_stop], verbose=1)

# Save final model
model.save("depression_sbert_optuna_model.keras")
print("Model saved as 'depression_sbert_optuna_model.keras'")
```

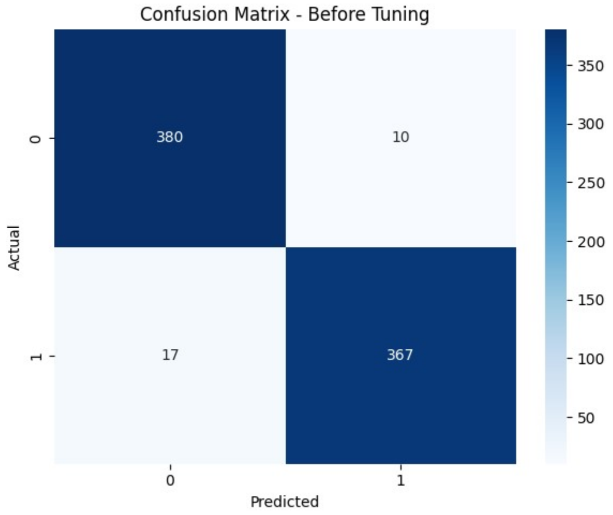
```
WARNING:absl:At this time, the v2.11+ optimizer
`tf.keras.optimizers.Adam` runs slowly on M1/M2 Macs, please use the
legacy Keras optimizer instead, located at
`tf.keras.optimizers.legacy.Adam`.
WARNING:absl:There is a known slowdown when using v2.11+ Keras
optimizers on M1/M2 Macs. Falling back to the legacy Keras optimizer,
i.e., `tf.keras.optimizers.legacy.Adam`.
```

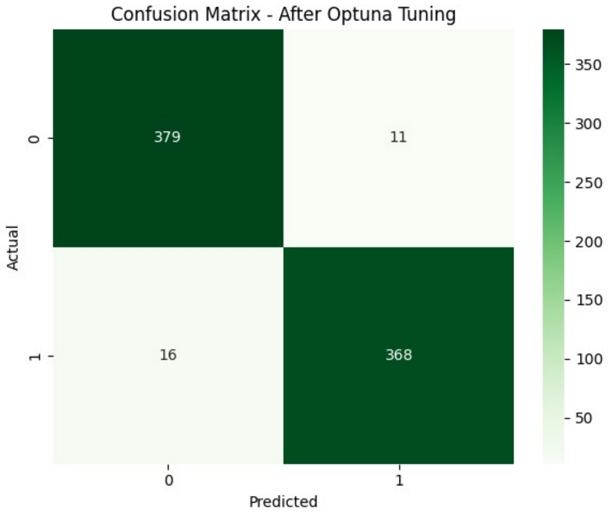
```
Epoch 1/25
194/194 [==============================] - 7s 27ms/step - loss: 0.5416
- accuracy: 0.9007 - val_loss: 0.7055 - val_accuracy: 0.9521
Epoch 2/25
194/194 [==============================] - 4s 21ms/step - loss: 0.4516
- accuracy: 0.9408 - val_loss: 0.4822 - val_accuracy: 0.9573
Epoch 3/25
194/194 [==============================] - 4s 21ms/step - loss: 0.4188
- accuracy: 0.9513 - val_loss: 0.3858 - val_accuracy: 0.9560
Epoch 4/25
194/194 [==============================] - 4s 21ms/step - loss: 0.4069
- accuracy: 0.9489 - val_loss: 0.3648 - val_accuracy: 0.9573
Epoch 5/25
194/194 [==============================] - 4s 21ms/step - loss: 0.3881
- accuracy: 0.9541 - val_loss: 0.3491 - val_accuracy: 0.9599
Epoch 6/25
194/194 [==============================] - 4s 21ms/step - loss: 0.3720
- accuracy: 0.9580 - val_loss: 0.3419 - val_accuracy: 0.9573
Epoch 7/25
194/194 [==============================] - 4s 21ms/step - loss: 0.3609
- accuracy: 0.9571 - val_loss: 0.3291 - val_accuracy: 0.9612
Epoch 8/25
194/194 [==============================] - 4s 21ms/step - loss: 0.3453
- accuracy: 0.9599 - val_loss: 0.3178 - val_accuracy: 0.9625
Epoch 9/25
194/194 [==============================] - 4s 21ms/step - loss: 0.3478
- accuracy: 0.9567 - val_loss: 0.3137 - val_accuracy: 0.9599
Epoch 10/25
194/194 [==============================] - 4s 21ms/step - loss: 0.3278
- accuracy: 0.9581 - val_loss: 0.3033 - val_accuracy: 0.9612
Epoch 11/25
194/194 [==============================] - 4s 21ms/step - loss: 0.3195
- accuracy: 0.9636 - val_loss: 0.2967 - val_accuracy: 0.9599
Epoch 12/25
194/194 [==============================] - 4s 21ms/step - loss: 0.3187
- accuracy: 0.9580 - val_loss: 0.2935 - val_accuracy: 0.9599
Epoch 13/25
194/194 [==============================] - 4s 21ms/step - loss: 0.3057
- accuracy: 0.9620 - val_loss: 0.2822 - val_accuracy: 0.9638
Epoch 14/25
194/194 [==============================] - 4s 21ms/step - loss: 0.3023
- accuracy: 0.9609 - val_loss: 0.2793 - val_accuracy: 0.9638
Epoch 15/25
194/194 [==============================] - 4s 21ms/step - loss: 0.2954
- accuracy: 0.9614 - val_loss: 0.2745 - val_accuracy: 0.9599
Epoch 16/25
194/194 [==============================] - 4s 21ms/step - loss: 0.2879
- accuracy: 0.9578 - val_loss: 0.2706 - val_accuracy: 0.9612
Epoch 17/25
194/194 [==============================] - 4s 21ms/step - loss: 0.2800
```

```
- accuracy: 0.9623 - val_loss: 0.2615 - val_accuracy: 0.9625
Epoch 18/25
194/194 [==============================] - 4s 21ms/step - loss: 0.2808
- accuracy: 0.9610 - val_loss: 0.2570 - val_accuracy: 0.9612
Epoch 19/25
194/194 [==============================] - 4s 21ms/step - loss: 0.2685
- accuracy: 0.9609 - val_loss: 0.2552 - val_accuracy: 0.9599
Epoch 20/25
194/194 [==============================] - 4s 21ms/step - loss: 0.2705
- accuracy: 0.9615 - val_loss: 0.2491 - val_accuracy: 0.9612
Epoch 21/25
194/194 [==============================] - 4s 21ms/step - loss: 0.2622
- accuracy: 0.9612 - val_loss: 0.2471 - val_accuracy: 0.9625
Epoch 22/25
194/194 [==============================] - 4s 21ms/step - loss: 0.2565
- accuracy: 0.9644 - val_loss: 0.2407 - val_accuracy: 0.9625
Epoch 23/25
194/194 [==============================] - 4s 21ms/step - loss: 0.2472
- accuracy: 0.9651 - val_loss: 0.2366 - val_accuracy: 0.9612
Epoch 24/25
194/194 [==============================] - 4s 21ms/step - loss: 0.2505
- accuracy: 0.9654 - val_loss: 0.2313 - val_accuracy: 0.9625
Epoch 25/25
194/194 [==============================] - 4s 21ms/step - loss: 0.2509
- accuracy: 0.9609 - val_loss: 0.2288 - val_accuracy: 0.9612
Model saved as 'depression_sbert_optuna_model.keras'

from tensorflow.keras.models import load_model
from sklearn.metrics import accuracy_score, precision_score,
recall_score, f1_score, roc_auc_score, confusion_matrix
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np

# Load models
baseline_model = load_model("depression_sbert_model.keras")
optuna_model = load_model("depression_sbert_optuna_model.keras")

# Predict on datasets
y_pred_baseline = (baseline_model.predict(X_test) > 0.5).astype(int)
y_pred_optuna = (optuna_model.predict(X_test) > 0.5).astype(int)

# Define function to compute all metrics
def get_metrics(y_true, y_pred):
    return {
        'Accuracy': round(accuracy_score(y_true, y_pred), 4),
        'Precision': round(precision_score(y_true, y_pred), 4),
        'Recall': round(recall_score(y_true, y_pred), 4),
        'F1 Score': round(f1_score(y_true, y_pred), 4),
        'ROC AUC': round(roc_auc_score(y_true, y_pred), 4)
    }

# Compute metrics
baseline_metrics = get_metrics(y_test, y_pred_baseline)
optuna_metrics = get_metrics(y_test, y_pred_optuna)

# Create comparison table
comparison_df = pd.DataFrame(
    [baseline_metrics, optuna_metrics],
    index=["Before Tuning", "After Optuna Tuning"]
)

# Print and save metrics
print("\nModel Performance Comparison:\n")
print(comparison_df)
comparison_df.to_csv("model_performance_comparison.csv")
print("\nSaved as 'model_performance_comparison.csv'")

# --- Bar Chart of Metrics ---
comparison_df.plot(kind='bar', figsize=(10, 6))
plt.title("Model Performance Comparison")
plt.ylabel("Score")
plt.ylim(0, 1)
plt.xticks(rotation=0)
plt.grid(True)
plt.tight_layout()
plt.show()

# --- Confusion Matrices ---
cm_baseline = confusion_matrix(y_test, y_pred_baseline)
cm_optuna = confusion_matrix(y_test, y_pred_optuna)

fig, axes = plt.subplots(1, 2, figsize=(12, 5))
sns.heatmap(cm_baseline, annot=True, fmt='d', cmap='Blues',
ax=axes[0])
axes[0].set_title("Confusion Matrix - Before Tuning")
axes[0].set_xlabel("Predicted")
axes[0].set_ylabel("Actual")

sns.heatmap(cm_optuna, annot=True, fmt='d', cmap='Greens', ax=axes[1])
axes[1].set_title("Confusion Matrix - After Optuna Tuning")
axes[1].set_xlabel("Predicted")
axes[1].set_ylabel("Actual")

plt.tight_layout()
plt.show()

# --- Accuracy on Train/Val/Test ---
train_acc_before = accuracy_score(y_train,
(baseline_model.predict(X_train) > 0.5).astype(int))
val_acc_before = accuracy_score(y_val, (baseline_model.predict(X_val)
> 0.5).astype(int))
test_acc_before = accuracy_score(y_test, y_pred_baseline)

train_acc_after = accuracy_score(y_train,
(optuna_model.predict(X_train) > 0.5).astype(int))
val_acc_after = accuracy_score(y_val, (optuna_model.predict(X_val) >
0.5).astype(int))
test_acc_after = accuracy_score(y_test, y_pred_optuna)

perf_data = {
    "Dataset": ["Train", "Validation", "Test"],
    "Before Tuning": [train_acc_before, val_acc_before,
test_acc_before],
    "After Optuna Tuning": [train_acc_after, val_acc_after,
test_acc_after]
}

perf_df = pd.DataFrame(perf_data)
perf_melted = perf_df.melt(id_vars="Dataset", var_name="Model",
value_name="Accuracy")

plt.figure(figsize=(8, 6))
sns.barplot(data=perf_melted, x="Dataset", y="Accuracy", hue="Model")
plt.title("Accuracy on Train, Validation, and Test Sets")
plt.ylim(0, 1)
plt.grid(axis='y')
plt.tight_layout()
plt.show()
```

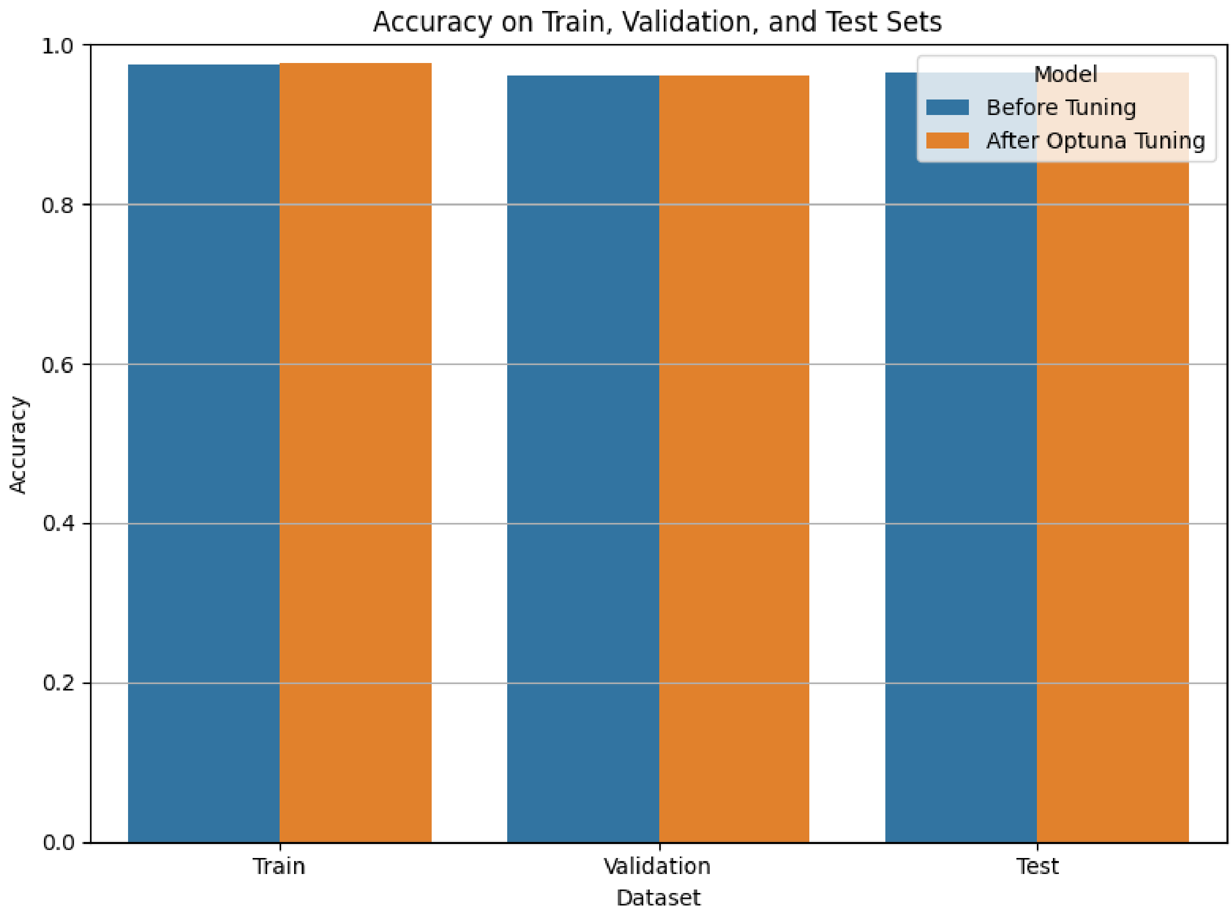
```
25/25 [==============================] - 0s 8ms/step
25/25 [==============================] - 0s 4ms/step

Model Performance Comparison:

                     Accuracy  Precision  Recall  F1 Score  ROC AUC
Before Tuning          0.9651     0.9735  0.9557    0.9645   0.9650
After Optuna Tuning    0.9651     0.9710  0.9583    0.9646   0.9651

Saved as 'model_performance_comparison.csv'
```







```
194/194 [==============================] - 1s 3ms/step
25/25 [==============================] - 0s 4ms/step
194/194 [==============================] - 0s 2ms/step
25/25 [==============================] - 0s 3ms/step
```



```python
import numpy as np
from tensorflow.keras.models import load_model

# Load Optuna-tuned model
model = load_model("depression_sbert_optuna_model.keras")

print("\n==== Real-time Depression Prediction ====")
print("Type a sentence to analyze or type 'exit' to quit.\n")

while True:
    user_input = input("Post: ").strip()
    if user_input.lower() == 'exit':
        print("Exiting prediction mode.")
        break
    if not user_input:
        print("Please enter some text.\n")
        continue

    # Preprocess and encode using SBERT
    cleaned = clean_text(user_input)
    embedding = sbert_model.encode([cleaned])
    # Predict
    prediction = model.predict(np.array(embedding))
    prob = prediction[0][0]
    label = "Depressed" if prob > 0.5 else "Not Depressed"
    confidence = round(prob * 100 if label == "Depressed" else (1 -
prob) * 100, 2)

    # Output with original input
    print("\nInput:", user_input)
    print(f"Prediction: {label} (Confidence: {confidence}%)\n")
```

```
==== Real-time Depression Prediction ====
Type a sentence to analyze or type 'exit' to quit.


/Library/Frameworks/Python.framework/Versions/3.10/lib/python3.10/
site-packages/torch/nn/modules/module.py:1762: FutureWarning:
`encoder_attention_mask` is deprecated and will be removed in version
4.55.0 for `BertSdpaSelfAttention.forward`.
  return forward_call(*args, **kwargs)

1/1 [==============================] - 0s 407ms/step

Input: I don’t see the point of anything anymore. I just want to sleep
and not wake up.
Prediction: Depressed (Confidence: 97.97%)

1/1 [==============================] - 0s 95ms/step

Input: I’m really excited about my new job – can’t wait to start next
week!
Prediction: Not Depressed (Confidence: 98.93%)

1/1 [==============================] - 0s 42ms/step

Input: Even when I’m surrounded by people, I feel completely alone and
numb.
Prediction: Depressed (Confidence: 99.68%)

1/1 [==============================] - 0s 35ms/step

Input: Had a great time catching up with old friends over dinner.
Prediction: Not Depressed (Confidence: 99.51%)

1/1 [==============================] - 0s 20ms/step

Input: It’s getting harder to get out of bed every day. I have no
motivation or energy left
Prediction: Not Depressed (Confidence: 68.64%)
```

```
1/1 [==============================] - 0s 51ms/step

Input: Just finished a long workout and I feel amazing. Ready to
tackle the day!
Prediction: Not Depressed (Confidence: 91.68%)

Please enter some text.
```