Install Required Libraries

```
!pip install datasets
!pip install kagglehub
!pip install transformers
!pip install evaluate
!pip install transformers datasets evaluate huggingface_hub -q
```

```
Collecting datasets
  Downloading datasets-3.5.0-py3-none-any.whl.metadata (19 kB)
Requirement already satisfied: filelock in
/usr/local/lib/python3.11/dist-packages (from datasets) (3.18.0)
Requirement already satisfied: numpy>=1.17 in
/usr/local/lib/python3.11/dist-packages (from datasets) (2.0.2)
Requirement already satisfied: pyarrow>=15.0.0 in
/usr/local/lib/python3.11/dist-packages (from datasets) (18.1.0)
Collecting dill<0.3.9,>=0.3.0 (from datasets)
  Downloading dill-0.3.8-py3-none-any.whl.metadata (10 kB)
Requirement already satisfied: pandas in
/usr/local/lib/python3.11/dist-packages (from datasets) (2.2.2)
Requirement already satisfied: requests>=2.32.2 in
/usr/local/lib/python3.11/dist-packages (from datasets) (2.32.3)
Requirement already satisfied: tqdm>=4.66.3 in
/usr/local/lib/python3.11/dist-packages (from datasets) (4.67.1)
Collecting xxhash (from datasets)
  Downloading xxhash-3.5.0-cp311-cp311-
manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (12 kB)
Collecting multiprocess<0.70.17 (from datasets)
  Downloading multiprocess-0.70.16-py311-none-any.whl.metadata (7.2
kB)
Collecting fsspec<=2024.12.0,>=2023.1.0 (from
fsspec[http]<=2024.12.0,>=2023.1.0->datasets)
  Downloading fsspec-2024.12.0-py3-none-any.whl.metadata (11 kB)
Requirement already satisfied: aiohttp in
/usr/local/lib/python3.11/dist-packages (from datasets) (3.11.15)
Requirement already satisfied: huggingface-hub>=0.24.0 in
/usr/local/lib/python3.11/dist-packages (from datasets) (0.30.2)
Requirement already satisfied: packaging in
/usr/local/lib/python3.11/dist-packages (from datasets) (24.2)
Requirement already satisfied: pyyaml>=5.1 in
/usr/local/lib/python3.11/dist-packages (from datasets) (6.0.2)
Requirement already satisfied: aiohappyeyeballs>=2.3.0 in
/usr/local/lib/python3.11/dist-packages (from aiohttp->datasets)
(2.6.1)
Requirement already satisfied: aiosignal>=1.1.2 in
/usr/local/lib/python3.11/dist-packages (from aiohttp->datasets)
(1.3.2)
Requirement already satisfied: attrs>=17.3.0 in
/usr/local/lib/python3.11/dist-packages (from aiohttp->datasets)
```

```
(25.3.0)
Requirement already satisfied: frozenlist>=1.1.1 in
/usr/local/lib/python3.11/dist-packages (from aiohttp->datasets)
(1.5.0)
Requirement already satisfied: multidict<7.0,>=4.5 in
/usr/local/lib/python3.11/dist-packages (from aiohttp->datasets)
(6.4.3)
Requirement already satisfied: propcache>=0.2.0 in
/usr/local/lib/python3.11/dist-packages (from aiohttp->datasets)
(0.3.1)
Requirement already satisfied: yarl<2.0,>=1.17.0 in
/usr/local/lib/python3.11/dist-packages (from aiohttp->datasets)
(1.19.0)
Requirement already satisfied: typing-extensions>=3.7.4.3 in
/usr/local/lib/python3.11/dist-packages (from huggingface-hub>=0.24.0-
>datasets) (4.13.2)
Requirement already satisfied: charset-normalizer<4,>=2 in
/usr/local/lib/python3.11/dist-packages (from requests>=2.32.2-
>datasets) (3.4.1)
Requirement already satisfied: idna<4,>=2.5 in
/usr/local/lib/python3.11/dist-packages (from requests>=2.32.2-
>datasets) (3.10)
Requirement already satisfied: urllib3<3,>=1.21.1 in
/usr/local/lib/python3.11/dist-packages (from requests>=2.32.2-
>datasets) (2.3.0)
Requirement already satisfied: certifi>=2017.4.17 in
/usr/local/lib/python3.11/dist-packages (from requests>=2.32.2-
>datasets) (2025.1.31)
Requirement already satisfied: python-dateutil>=2.8.2 in
/usr/local/lib/python3.11/dist-packages (from pandas->datasets)
(2.8.2)
Requirement already satisfied: pytz>=2020.1 in
/usr/local/lib/python3.11/dist-packages (from pandas->datasets)
(2025.2)
Requirement already satisfied: tzdata>=2022.7 in
/usr/local/lib/python3.11/dist-packages (from pandas->datasets)
(2025.2)
Requirement already satisfied: six>=1.5 in
/usr/local/lib/python3.11/dist-packages (from python-dateutil>=2.8.2-
>pandas->datasets) (1.17.0)
Downloading datasets-3.5.0-py3-none-any.whl (491 kB)
━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 491.2/491.2 kB 7.4 MB/s eta
0:00:00
━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 116.3/116.3 kB 6.9 MB/s eta
0:00:00
━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 183.9/183.9 kB 8.0 MB/s eta
0:00:00
ultiprocess-0.70.16-py311-none-any.whl (143 kB)
━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 143.5/143.5 kB 4.0 MB/s eta
```

```
0:00:00
anylinux_2_17_x86_64.manylinux2014_x86_64.whl (194 kB)
━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 194.8/194.8 kB 7.7 MB/s eta
0:00:00
ultiprocess, datasets
  Attempting uninstall: fsspec
    Found existing installation: fsspec 2025.3.2
    Uninstalling fsspec-2025.3.2:
      Successfully uninstalled fsspec-2025.3.2
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.
torch 2.6.0+cu124 requires nvidia-cublas-cu12==12.4.5.8;
platform_system == "Linux" and platform_machine == "x86_64", but you
have nvidia-cublas-cu12 12.5.3.2 which is incompatible.
torch 2.6.0+cu124 requires nvidia-cuda-cupti-cu12==12.4.127;
platform_system == "Linux" and platform_machine == "x86_64", but you
have nvidia-cuda-cupti-cu12 12.5.82 which is incompatible.
torch 2.6.0+cu124 requires nvidia-cuda-nvrtc-cu12==12.4.127;
platform_system == "Linux" and platform_machine == "x86_64", but you
have nvidia-cuda-nvrtc-cu12 12.5.82 which is incompatible.
torch 2.6.0+cu124 requires nvidia-cuda-runtime-cu12==12.4.127;
platform_system == "Linux" and platform_machine == "x86_64", but you
have nvidia-cuda-runtime-cu12 12.5.82 which is incompatible.
torch 2.6.0+cu124 requires nvidia-cudnn-cu12==9.1.0.70;
platform_system == "Linux" and platform_machine == "x86_64", but you
have nvidia-cudnn-cu12 9.3.0.75 which is incompatible.
torch 2.6.0+cu124 requires nvidia-cufft-cu12==11.2.1.3;
platform_system == "Linux" and platform_machine == "x86_64", but you
have nvidia-cufft-cu12 11.2.3.61 which is incompatible.
torch 2.6.0+cu124 requires nvidia-curand-cu12==10.3.5.147;
platform_system == "Linux" and platform_machine == "x86_64", but you
have nvidia-curand-cu12 10.3.6.82 which is incompatible.
torch 2.6.0+cu124 requires nvidia-cusolver-cu12==11.6.1.9;
platform_system == "Linux" and platform_machine == "x86_64", but you
have nvidia-cusolver-cu12 11.6.3.83 which is incompatible.
torch 2.6.0+cu124 requires nvidia-cusparse-cu12==12.3.1.170;
platform_system == "Linux" and platform_machine == "x86_64", but you
have nvidia-cusparse-cu12 12.5.1.3 which is incompatible.
torch 2.6.0+cu124 requires nvidia-nvjitlink-cu12==12.4.127;
platform_system == "Linux" and platform_machine == "x86_64", but you
have nvidia-nvjitlink-cu12 12.5.82 which is incompatible.
gcsfs 2025.3.2 requires fsspec==2025.3.2, but you have fsspec
2024.12.0 which is incompatible.
Successfully installed datasets-3.5.0 dill-0.3.8 fsspec-2024.12.0
multiprocess-0.70.16 xxhash-3.5.0
Requirement already satisfied: kagglehub in
/usr/local/lib/python3.11/dist-packages (0.3.11)
Requirement already satisfied: packaging in
```

```
/usr/local/lib/python3.11/dist-packages (from kagglehub) (24.2)
Requirement already satisfied: pyyaml in
/usr/local/lib/python3.11/dist-packages (from kagglehub) (6.0.2)
Requirement already satisfied: requests in
/usr/local/lib/python3.11/dist-packages (from kagglehub) (2.32.3)
Requirement already satisfied: tqdm in /usr/local/lib/python3.11/dist-
packages (from kagglehub) (4.67.1)
Requirement already satisfied: charset-normalizer<4,>=2 in
/usr/local/lib/python3.11/dist-packages (from requests->kagglehub)
(3.4.1)
Requirement already satisfied: idna<4,>=2.5 in
/usr/local/lib/python3.11/dist-packages (from requests->kagglehub)
(3.10)
Requirement already satisfied: urllib3<3,>=1.21.1 in
/usr/local/lib/python3.11/dist-packages (from requests->kagglehub)
(2.3.0)
Requirement already satisfied: certifi>=2017.4.17 in
/usr/local/lib/python3.11/dist-packages (from requests->kagglehub)
(2025.1.31)
Requirement already satisfied: transformers in
/usr/local/lib/python3.11/dist-packages (4.51.3)
Requirement already satisfied: filelock in
/usr/local/lib/python3.11/dist-packages (from transformers) (3.18.0)
Requirement already satisfied: huggingface-hub<1.0,>=0.30.0 in
/usr/local/lib/python3.11/dist-packages (from transformers) (0.30.2)
Requirement already satisfied: numpy>=1.17 in
/usr/local/lib/python3.11/dist-packages (from transformers) (2.0.2)
Requirement already satisfied: packaging>=20.0 in
/usr/local/lib/python3.11/dist-packages (from transformers) (24.2)
Requirement already satisfied: pyyaml>=5.1 in
/usr/local/lib/python3.11/dist-packages (from transformers) (6.0.2)
Requirement already satisfied: regex!=2019.12.17 in
/usr/local/lib/python3.11/dist-packages (from transformers)
(2024.11.6)
Requirement already satisfied: requests in
/usr/local/lib/python3.11/dist-packages (from transformers) (2.32.3)
Requirement already satisfied: tokenizers<0.22,>=0.21 in
/usr/local/lib/python3.11/dist-packages (from transformers) (0.21.1)
Requirement already satisfied: safetensors>=0.4.3 in
/usr/local/lib/python3.11/dist-packages (from transformers) (0.5.3)
Requirement already satisfied: tqdm>=4.27 in
/usr/local/lib/python3.11/dist-packages (from transformers) (4.67.1)
Requirement already satisfied: fsspec>=2023.5.0 in
/usr/local/lib/python3.11/dist-packages (from huggingface-
hub<1.0,>=0.30.0->transformers) (2024.12.0)
Requirement already satisfied: typing-extensions>=3.7.4.3 in
/usr/local/lib/python3.11/dist-packages (from huggingface-
hub<1.0,>=0.30.0->transformers) (4.13.2)
Requirement already satisfied: charset-normalizer<4,>=2 in
```

```
/usr/local/lib/python3.11/dist-packages (from requests->transformers)
(3.4.1)
Requirement already satisfied: idna<4,>=2.5 in
/usr/local/lib/python3.11/dist-packages (from requests->transformers)
(3.10)
Requirement already satisfied: urllib3<3,>=1.21.1 in
/usr/local/lib/python3.11/dist-packages (from requests->transformers)
(2.3.0)
Requirement already satisfied: certifi>=2017.4.17 in
/usr/local/lib/python3.11/dist-packages (from requests->transformers)
(2025.1.31)
Collecting evaluate
  Downloading evaluate-0.4.3-py3-none-any.whl.metadata (9.2 kB)
Requirement already satisfied: datasets>=2.0.0 in
/usr/local/lib/python3.11/dist-packages (from evaluate) (3.5.0)
Requirement already satisfied: numpy>=1.17 in
/usr/local/lib/python3.11/dist-packages (from evaluate) (2.0.2)
Requirement already satisfied: dill in /usr/local/lib/python3.11/dist-
packages (from evaluate) (0.3.8)
Requirement already satisfied: pandas in
/usr/local/lib/python3.11/dist-packages (from evaluate) (2.2.2)
Requirement already satisfied: requests>=2.19.0 in
/usr/local/lib/python3.11/dist-packages (from evaluate) (2.32.3)
Requirement already satisfied: tqdm>=4.62.1 in
/usr/local/lib/python3.11/dist-packages (from evaluate) (4.67.1)
Requirement already satisfied: xxhash in
/usr/local/lib/python3.11/dist-packages (from evaluate) (3.5.0)
Requirement already satisfied: multiprocess in
/usr/local/lib/python3.11/dist-packages (from evaluate) (0.70.16)
Requirement already satisfied: fsspec>=2021.05.0 in
/usr/local/lib/python3.11/dist-packages (from fsspec[http]>=2021.05.0-
>evaluate) (2024.12.0)
Requirement already satisfied: huggingface-hub>=0.7.0 in
/usr/local/lib/python3.11/dist-packages (from evaluate) (0.30.2)
Requirement already satisfied: packaging in
/usr/local/lib/python3.11/dist-packages (from evaluate) (24.2)
Requirement already satisfied: filelock in
/usr/local/lib/python3.11/dist-packages (from datasets>=2.0.0-
>evaluate) (3.18.0)
Requirement already satisfied: pyarrow>=15.0.0 in
/usr/local/lib/python3.11/dist-packages (from datasets>=2.0.0-
>evaluate) (18.1.0)
Requirement already satisfied: aiohttp in
/usr/local/lib/python3.11/dist-packages (from datasets>=2.0.0-
>evaluate) (3.11.15)
Requirement already satisfied: pyyaml>=5.1 in
/usr/local/lib/python3.11/dist-packages (from datasets>=2.0.0-
>evaluate) (6.0.2)
Requirement already satisfied: typing-extensions>=3.7.4.3 in
```

```
/usr/local/lib/python3.11/dist-packages (from huggingface-hub>=0.7.0-
>evaluate) (4.13.2)
Requirement already satisfied: charset-normalizer<4,>=2 in
/usr/local/lib/python3.11/dist-packages (from requests>=2.19.0-
>evaluate) (3.4.1)
Requirement already satisfied: idna<4,>=2.5 in
/usr/local/lib/python3.11/dist-packages (from requests>=2.19.0-
>evaluate) (3.10)
Requirement already satisfied: urllib3<3,>=1.21.1 in
/usr/local/lib/python3.11/dist-packages (from requests>=2.19.0-
>evaluate) (2.3.0)
Requirement already satisfied: certifi>=2017.4.17 in
/usr/local/lib/python3.11/dist-packages (from requests>=2.19.0-
>evaluate) (2025.1.31)
Requirement already satisfied: python-dateutil>=2.8.2 in
/usr/local/lib/python3.11/dist-packages (from pandas->evaluate)
(2.8.2)
Requirement already satisfied: pytz>=2020.1 in
/usr/local/lib/python3.11/dist-packages (from pandas->evaluate)
(2025.2)
Requirement already satisfied: tzdata>=2022.7 in
/usr/local/lib/python3.11/dist-packages (from pandas->evaluate)
(2025.2)
Requirement already satisfied: aiohappyeyeballs>=2.3.0 in
/usr/local/lib/python3.11/dist-packages (from aiohttp-
>datasets>=2.0.0->evaluate) (2.6.1)
Requirement already satisfied: aiosignal>=1.1.2 in
/usr/local/lib/python3.11/dist-packages (from aiohttp-
>datasets>=2.0.0->evaluate) (1.3.2)
Requirement already satisfied: attrs>=17.3.0 in
/usr/local/lib/python3.11/dist-packages (from aiohttp-
>datasets>=2.0.0->evaluate) (25.3.0)
Requirement already satisfied: frozenlist>=1.1.1 in
/usr/local/lib/python3.11/dist-packages (from aiohttp-
>datasets>=2.0.0->evaluate) (1.5.0)
Requirement already satisfied: multidict<7.0,>=4.5 in
/usr/local/lib/python3.11/dist-packages (from aiohttp-
>datasets>=2.0.0->evaluate) (6.4.3)
Requirement already satisfied: propcache>=0.2.0 in
/usr/local/lib/python3.11/dist-packages (from aiohttp-
>datasets>=2.0.0->evaluate) (0.3.1)
Requirement already satisfied: yarl<2.0,>=1.17.0 in
/usr/local/lib/python3.11/dist-packages (from aiohttp-
>datasets>=2.0.0->evaluate) (1.19.0)
Requirement already satisfied: six>=1.5 in
/usr/local/lib/python3.11/dist-packages (from python-dateutil>=2.8.2-
>pandas->evaluate) (1.17.0)
Downloading evaluate-0.4.3-py3-none-any.whl (84 kB)
```

```
━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 84.0/84.0 kB 2.1 MB/s eta
0:00:00
```

Importing Required Librires

```
import os
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import sklearn
import kagglehub
import evaluate
import torch as t
import transformers
from sklearn.metrics import accuracy_score, f1_score
from transformers import
AutoModelForSequenceClassification,AutoTokenizer,DataCollatorWithPaddi
ng,Trainer,TrainingArguments, EarlyStoppingCallback
from datasets import
load_dataset,DatasetDict,ClassLabel,Features,Value

t.cuda.is_available()
```

```
True
```

Installing pre_trined Model

```
tokenizer = AutoTokenizer.from_pretrained('google-bert/bert-base-
multilingual-cased')
```

```
/usr/local/lib/python3.11/dist-packages/huggingface_hub/utils/
_auth.py:94: UserWarning:
The secret `HF_TOKEN` does not exist in your Colab secrets.
To authenticate with the Hugging Face Hub, create a token in your
settings tab (https://huggingface.co/settings/tokens), set it as
secret in your Google Colab and restart your session.
You will be able to reuse this secret in all of your notebooks.
Please note that authentication is recommended but still optional to
access public models or datasets.
  warnings.warn(
```


```
{"model_id":"9aa8a850bb024bf68240f442c13ac710","version_major":2,"vers
ion_minor":0}
```

```
{"model_id":"73d65631d5f947eaa0325e548b28f41d","version_major":2,"vers
ion_minor":0}
```

```
{"model_id":"3a0de6cf07c94e2ba8aa1af4073b508b","version_major":2,"vers
ion_minor":0}
```

{"model_id":"bdcc43433f984c31a00b6588d9bafe98","version_major":2,"version_minor":0}

```
tokenizer
```

```
BertTokenizerFast(name_or_path='google-bert/bert-base-multilingual-cased', vocab_size=119547, model_max_length=512, is_fast=True, padding_side='right', truncation_side='right', special_tokens={'unk_token': '[UNK]', 'sep_token': '[SEP]', 'pad_token': '[PAD]', 'cls_token': '[CLS]', 'mask_token': '[MASK]'}, clean_up_tokenization_spaces=False, added_tokens_decoder={
	0: AddedToken("[PAD]", rstrip=False, lstrip=False, single_word=False, normalized=False, special=True),
	100: AddedToken("[UNK]", rstrip=False, lstrip=False, single_word=False, normalized=False, special=True),
	101: AddedToken("[CLS]", rstrip=False, lstrip=False, single_word=False, normalized=False, special=True),
	102: AddedToken("[SEP]", rstrip=False, lstrip=False, single_word=False, normalized=False, special=True),
	103: AddedToken("[MASK]", rstrip=False, lstrip=False, single_word=False, normalized=False, special=True),
}
)
```

lodaing data into dict

```
data=load_dataset('Sanath369/Telugu_sentiment_sentences')
```

{"model_id":"d1dc2bd78271433296a556fa17eeafde","version_major":2,"version_minor":0}

{"model_id":"32282f83ff0947de9c1ac801abb69ecd","version_major":2,"version_minor":0}

{"model_id":"0f1529d159984149831c45325c5b89aa","version_major":2,"version_minor":0}

{"model_id":"19e809df17e14bb2a1f25d5b8a7bf5d3","version_major":2,"version_minor":0}

{"model_id":"c33ce10cae24492ca64310648cda2b91","version_major":2,"version_minor":0}

```
data
```

```
DatasetDict({
    train: Dataset({
        features: ['text', 'label'],
        num_rows: 24599
    })
    test: Dataset({
```

```
        features: ['text', 'label'],
        num_rows: 7033
    })
})
```

```
#split the train data into train and Validation
main_data1=data['train'].train_test_split(test_size=0.2)
```

```
main_data1
```

```
DatasetDict({
    train: Dataset({
        features: ['text', 'label'],
        num_rows: 19679
    })
    test: Dataset({
        features: ['text', 'label'],
        num_rows: 4920
    })
})
```

```
# creating the final dataset
final_data={'train':main_data1['train'],'test':data['test'],'validatio
n':main_data1['test']}
```

```
final_data
```

```
{'train': Dataset({
     features: ['text', 'label'],
     num_rows: 19679
 }),
 'test': Dataset({
     features: ['text', 'label'],
     num_rows: 7033
 }),
 'validation': Dataset({
     features: ['text', 'label'],
     num_rows: 4920
 })}
```

Prepare data for Model Input

```
#Text Preprocessing: Handling Empty Reviews & Tokenization
def pre_processing(batch):
    review_text = []
    for i in batch['text']:
        if i and i.strip():
            review_text.append(i)
        else:
            review_text.append("[EMPTY]")
    tokenizers = tokenizer(
```

```
        review_text,
        truncation=True,
        padding='max_length',
        max_length=512
    )
    return tokenizers

final_data = DatasetDict(final_data)

#passing the preproccing funtion
final_tokenized_data=final_data.map(pre_processing,batched=True,remove
_columns=['text'])

{"model_id":"c7497f659054448299f0e92038640674","version_major":2,"vers
ion_minor":0}

{"model_id":"1859f939c71d4076a32101577a9792b5","version_major":2,"vers
ion_minor":0}

{"model_id":"8878f8c8f2c14343928c7076163836d7","version_major":2,"vers
ion_minor":0}

final_tokenized_data

DatasetDict({
    train: Dataset({
        features: ['label', 'input_ids', 'token_type_ids',
'attention_mask'],
        num_rows: 19679
    })
    test: Dataset({
        features: ['label', 'input_ids', 'token_type_ids',
'attention_mask'],
        num_rows: 7033
    })
    validation: Dataset({
        features: ['label', 'input_ids', 'token_type_ids',
'attention_mask'],
        num_rows: 4920
    })
})
```

Load Pretrained BERT Model for Sequence Classification

```
model=AutoModelForSequenceClassification.from_pretrained('google-
bert/bert-base-multilingual-cased', num_labels=3)

Xet Storage is enabled for this repo, but the 'hf_xet' package is not
installed. Falling back to regular HTTP download. For better
performance, install the package with: `pip install
huggingface_hub[hf_xet]` or `pip install hf_xet`
```

```
WARNING:huggingface_hub.file_download:Xet Storage is enabled for this repo, but the 'hf_xet' package is not installed. Falling back to regular HTTP download. For better performance, install the package with: `pip install huggingface_hub[hf_xet]` or `pip install hf_xet`
```


```
{"model_id":"e5f5b84b1c724b859f07c305466f2420","version_major":2,"version_minor":0}
```


```
Some weights of BertForSequenceClassification were not initialized from the model checkpoint at google-bert/bert-base-multilingual-cased and are newly initialized: ['classifier.bias', 'classifier.weight']
You should probably TRAIN this model on a down-stream task to be able to use it for predictions and inference.
```

Define Custom Metrics for Trainer Evaluation

```
def compute_metrics(eval_pred):
    # Unpack logits and labels
    logits, labels = eval_pred

    # Convert logits to NumPy if not already
    if not isinstance(logits, np.ndarray):
        logits = np.array(logits)

    # Convert labels to NumPy if not already
    if not isinstance(labels, np.ndarray):
        labels = np.array(labels)

    # Compute predictions by taking argmax along the last axis
    predictions = np.argmax(logits, axis=-1)

    # Compute accuracy
    acc = accuracy_score(labels, predictions)

    # Compute F1-score for 3-class sentiment analysis (weighted for imbalanced classes)
    f1 = f1_score(labels, predictions, average="weighted")  # Use "macro" for equal class weighting

    return {"accuracy": acc, "f1": f1}
    predictions = logits.argmax(axis=-1)
    acc = accuracy_score(labels, predictions)
    f1 = f1_score(labels, predictions, average="weighted")  # or "macro" / "binary"
    return {"accuracy": acc, "f1": f1}
```

Define TrainingArguments for Model Fine-Tuning

```
ta=TrainingArguments(eval_strategy ="epoch",
                     learning_rate=2e-5,save_strategy="epoch",
```

```
                  logging_dir="/content/.config",
                  num_train_epochs=5,output_dir="/content/sentiment-
bert-model",

per_device_train_batch_size=16,per_device_eval_batch_size=16,fp16=True
,
                  metric_for_best_model="eval_loss")
```

Trainer Setup for Fine-Tuning

```
tr=Trainer(model=model,
        args=ta,
        train_dataset=final_tokenized_data["train"],
        eval_dataset=final_tokenized_data["validation"],
        tokenizer=tokenizer,
        compute_metrics=compute_metrics,
        callbacks=[
        EarlyStoppingCallback(
            early_stopping_patience=2,       # stop after 3 epochs of
no real improvement
            early_stopping_threshold=0.01)])
```

```
<ipython-input-20-e1c629b8eee0>:1: FutureWarning: `tokenizer` is
deprecated and will be removed in version 5.0.0 for
`Trainer.__init__`. Use `processing_class` instead.
  tr=Trainer(model=model,
```

Start Model Training

```
tr.train()
```

```
wandb: WARNING The `run_name` is currently set to the same value as
`TrainingArguments.output_dir`. If this was not intended, please
specify a different run name by setting the
`TrainingArguments.run_name` parameter.
wandb: Using wandb-core as the SDK backend.  Please refer to
https://wandb.me/wandb-core for more information.

<IPython.core.display.Javascript object>

wandb: Logging into wandb.ai. (Learn how to deploy a W&B server
locally: https://wandb.me/wandb-server)
wandb: You can find your API key in your browser here:
https://wandb.ai/authorize
wandb: Paste an API key from your profile and hit enter:

 ..........

wandb: WARNING If you're specifying your api key in code, ensure this
code is not shared publicly.
wandb: WARNING Consider setting the WANDB_API_KEY environment
```

```
variable, or running `wandb login` from the command line.
wandb: No netrc file found, creating one.
wandb: Appending key for api.wandb.ai to your netrc file: /root/.netrc
wandb: Currently logged in as: pavanmuthyala45 (pavanmuthyala45-
innomatics-research-labs) to https://api.wandb.ai. Use `wandb login --
relogin` to force relogin
```

```
<IPython.core.display.HTML object>
```

```
<IPython.core.display.HTML object>
```

```
<IPython.core.display.HTML object>
```

```
<IPython.core.display.HTML object>
```

```
<IPython.core.display.HTML object>
```

```
Using EarlyStoppingCallback without load_best_model_at_end=True. Once
training is finished, the best model will not be loaded automatically.
```

```
<IPython.core.display.HTML object>
```

```
TrainOutput(global_step=4920, training_loss=0.6612224547843623,
metrics={'train_runtime': 2453.4868, 'train_samples_per_second':
40.104, 'train_steps_per_second': 2.507, 'total_flos':
2.07112357898772 5e+16, 'train_loss': 0.6612224547843623, 'epoch':
4.0})
```

```
results = tr.evaluate(eval_dataset=final_tokenized_data['test'])
print("Test Evaluation Results:", results)
```

```
<IPython.core.display.HTML object>
```

```
Test Evaluation Results: {'eval_loss': 0.8279711008071899,
'eval_accuracy': 0.6729702829517986, 'eval_f1': 0.6724756557429611,
'eval_runtime': 56.0613, 'eval_samples_per_second': 125.452,
'eval_steps_per_second': 7.849, 'epoch': 4.0}
```

Saving the model

```
tr.model.save_pretrained("/content/telugu_sentiment_bert_model_new")
tokenizer.save_pretrained("/content/telugu_sentiment_bert_model_new")
```

```
('/content/telugu_sentiment_bert_model_new/tokenizer_config.json',
 '/content/telugu_sentiment_bert_model_new/special_tokens_map.json',
 '/content/telugu_sentiment_bert_model_new/vocab.txt',
 '/content/telugu_sentiment_bert_model_new/added_tokens.json',
 '/content/telugu_sentiment_bert_model_new/tokenizer.json')
```

Loading Finetuned Model

```
from transformers import AutoModelForSequenceClassification,
AutoTokenizer

model =
AutoModelForSequenceClassification.from_pretrained("/content/telugu_se
ntiment_bert_model_new")
tokenizer =
AutoTokenizer.from_pretrained("/content/telugu_sentiment_bert_model_ne
w")
```

Steps to Upload Model to Hugging Face

1. Create a Model Repository on Hugging Face

- Visit Hugging Face, log in, and create a new model repository via your profile.
- Set the repository name and visibility (Public or Private) as desired.

2. Obtain an API Key from Hugging Face

- Go to your Hugging Face account settings.
- Select the option to create a new API key with **write** permissions.
- Copy and securely store the generated API token.

3. Install the Hugging Face Hub Library

- Install the `huggingface_hub` package to interact with Hugging Face. ```bash pip install huggingface_hub

4. Log in to Hugging Face in Your Environment

- Use the notebook_login function to authenticate in a Jupyter or Colab notebook.

```
from huggingface_hub import notebook_login
notebook_login()
```

- A pop-up will appear in Colab/Jupyter to enter your Hugging Face API token
- Enter the API token obtained in Step 2.

5. Save and Name Your Model

- Save your model and tokenizer locally, ensuring the name matches the repository name created on Hugging Face.

```
 from transformers import AutoTokenizer,
AutoModelForSequenceClassification
```

```
model.save_pretrained("your_model_directory")
tokenizer.save_pretrained("your_model_directory")
```

6. Push Your Model to Hugging Face

- Use the Hugging Face API to upload your model and tokenizer to the repository.

```
from huggingface_hub import HfApi, HfFolder, Repository
from transformers import AutoTokenizer,
AutoModelForSequenceClassification

# Push model and tokenizer to the repository
model.push_to_hub("username/your-model-name")
tokenizer.push_to_hub("username/your-model-name")
```

7. Create a Pipeline to Test the Model

- Use the pipeline helper to load and test your uploaded model. python

```
from transformers import pipeline

pipe = pipeline("text-classification", model="username/your-
model-name")
```

8. Verify the Model Upload

- Visit your model repository on Hugging Face (e.g., https://huggingface.co/username/your-model-name) to confirm the model and tokenizer were uploaded successfully.
- Test the pipeline to ensure the model works as expected.

Uploading model to hugging face Model

```
from huggingface_hub import notebook_login

notebook_login()  # A pop-up will appear in Colab asking for your HF
tokena


{"model_id":"a19c5fcd954c405ba2ee2d80b6257b58","version_major":2,"vers
ion_minor":0}

from huggingface_hub import HfApi, HfFolder, Repository
from transformers import AutoTokenizer,
AutoModelForSequenceClassification

# Push model and tokenizer
model.push_to_hub("Mpavan45/Telugu_Sentimental_Analysis")
tokenizer.push_to_hub("Mpavan45/Telugu_Sentimental_Analysis")a
```

{"model_id":"bd80600154ee4bc397ed37f6ded5a683","version_major":2,"version_minor":0}

{"model_id":"86ab56560c4447cbbec98fdd947b46b8","version_major":2,"version_minor":0}

{"type":"string"}

Importing finetuned model and creating pipe-line

```
# Use a pipeline as a high-level helper
from transformers import pipeline

pipe = pipeline("text-classification",
model="Mpavan45/Telugu_Sentimental_Analysis")
```

{"model_id":"17fe28720c784d438876e5d11cd35a2b","version_major":2,"version_minor":0}

{"model_id":"631bf60de04746c6b27a928356b86eac","version_major":2,"version_minor":0}

{"model_id":"9f9519b584d541cc91e6748ea3be3034","version_major":2,"version_minor":0}

{"model_id":"93188c7f122b481695cd2453443f3e60","version_major":2,"version_minor":0}

{"model_id":"53d0cb2ad01a4b73bf9e70bcb12d52ab","version_major":2,"version_minor":0}

{"model_id":"1f4adfdc0ec14f89ae7ba17b7311a9ee","version_major":2,"version_minor":0}

```
Device set to use cuda:0
```

Testing the model with different Telugu texts

```
# Example usage with different Telugu texts:
test_texts = [
     "□ □□□□□□ □□□□ □□□□□□□□",  # This movie is very good
    "□□□□ □ □□□□□□□□□□□□ □□□□ □□□□□□□□□□□□",  # I liked this book very
much
     "□ □□□□□ □□□□ □□□□□□ □□□□",  # This food is very bad
    "□□□□ □ □□□□ □□□□ □□□□□□□□ □□□□",  # I am very happy today
     "□□□□□ □ □□□□□□□□ □□□□ □□□□□□□□□□□",# I felt very sad for this
news
     "□ □□□□□□□□ □□□□ □□□□□□□□□ □□□□",# This weather is very
wonderful
     "□□□□□ □ □□□□□ □□□□□□ □□□□□□□□□□□□□□□□□",# I can not complete
this work
```

```
    "□ □□□□ □□□□ □□□□□□ □□□□",  # This car is very fast
    "□ □□□ □□□□ □□□□□□ □□□□",  # This song is very beautiful
    "□ □□□□□□□ □□□□ □□□□□□□□□□ □□□□", # This place is very calm
   "□□□□ □□□□ □□□□□□□□□□", # I am very tired
    "□ □□□□□□□ □□□□□□□□□ □□□□", # There is no solution to this
problem
    "□□□□ □ □□□□□□□□□□□ □□□□□□□□□□□□□□□□□□", # I can not believe this
information
    "□ □□□□□□□ □□□□ □□□□□□□□", # This person is very good
    "□ □□□□□ □□□□□□ □□□□□□ □□□□□□" #This work should be completed
quickly
    # # Add more examples here...
]

for text in test_texts:
    result = pipe(text)
    print(f"Text: {text}")
    print(f"Result: {result}")
    print("-" * 20)
```

```
You seem to be using the pipelines sequentially on GPU. In order to
maximize efficiency please use a dataset

Text: □ □□□□□□ □□□□ □□□□□□□
Result: [{'label': 'LABEL_2', 'score': 0.9709675312042236}]
--------------------
Text: □□□□ □ □□□□□□□□□□ □□□□ □□□□□□□□□□
Result: [{'label': 'LABEL_2', 'score': 0.9643860459327698}]
--------------------
Text: □ □□□□□ □□□□ □□□□□□ □□□□
Result: [{'label': 'LABEL_0', 'score': 0.4701785445213318}]
--------------------
Text: □□□□ □ □□□□ □□□□ □□□□□□□□ □□□□
Result: [{'label': 'LABEL_2', 'score': 0.9663376212120056}]
--------------------
Text: □□□□ □ □□□□□□□□ □□□□ □□□□□□□□□□
Result: [{'label': 'LABEL_0', 'score': 0.9451944231987}]
--------------------
Text: □ □□□□□□□□ □□□□ □□□□□□□□□ □□□□
Result: [{'label': 'LABEL_2', 'score': 0.9697619676589966}]
--------------------
Text: □□□□ □ □□□□□ □□□□□□ □□□□□□□□□□□□□□□□□
Result: [{'label': 'LABEL_1', 'score': 0.6042336821556091}]
--------------------
Text: □ □□□□ □□□□ □□□□□□ □□□□
Result: [{'label': 'LABEL_2', 'score': 0.8727412223815918}]
--------------------
Text: □ □□□ □□□□ □□□□□□ □□□□
```

```
Result: [{'label': 'LABEL_2', 'score': 0.9699376821517944}]
--------------------
Text: □ □□□□□□□□ □□□□□ □□□□□□□□□□□□ □□□□□
Result: [{'label': 'LABEL_2', 'score': 0.9308358430862427}]
--------------------
Text: □□□□□ □□□□□ □□□□□□□□□□□□
Result: [{'label': 'LABEL_2', 'score': 0.4581573009490967}]
--------------------
Text: □ □□□□□□□□ □□□□□□□□□□ □□□□□
Result: [{'label': 'LABEL_1', 'score': 0.5445381999015808}]
--------------------
Text: □□□□□ □ □□□□□□□□□□□□□□ □□□□□□□□□□□□□□□□□□□□
Result: [{'label': 'LABEL_1', 'score': 0.49418455362319946}]
--------------------
Text: □ □□□□□□□□ □□□□□ □□□□□□□□□□
Result: [{'label': 'LABEL_2', 'score': 0.9530943036079407}]
--------------------
Text: □ □□□□□□ □□□□□□□ □□□□□□□ □□□□□□□
Result: [{'label': 'LABEL_1', 'score': 0.7783124446868896}]
--------------------
```