DistilBERT

April 28, 2025

```
[ ]: ! pip install datasets
```

```
Collecting datasets
  Downloading datasets-3.5.1-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<=2025.3.0,>=2023.1.0 (from
fsspec[http]<=2025.3.0,>=2023.1.0->datasets)
  Downloading fsspec-2025.3.0-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) (1.6.0)
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) (1.20.0)
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)
(2.4.0)
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) (2.9.0.post0)
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-3.5.1-py3-none-any.whl (491 kB)
                          491.4/491.4 kB
9.0 MB/s eta 0:00:00
Downloading dill-0.3.8-py3-none-any.whl (116 kB)
                          116.3/116.3 kB
9.9 MB/s eta 0:00:00
Downloading fsspec-2025.3.0-py3-none-any.whl (193 kB)
                          193.6/193.6 kB
15.5 MB/s eta 0:00:00
Downloading multiprocess-0.70.16-py311-none-any.whl (143 kB)
                          143.5/143.5 kB
11.1 MB/s eta 0:00:00
Downloading
xxhash-3.5.0-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (194 kB)
                          194.8/194.8 kB
```

```
11.5 MB/s eta 0:00:00
Installing collected packages: xxhash, fsspec, dill, multiprocess,
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 2025.3.0 which is
incompatible.
Successfully installed datasets-3.5.1 dill-0.3.8 fsspec-2025.3.0
```

```
multiprocess-0.70.16 xxhash-3.5.0
```

[ ]:

```
from transformers import DistilBertTokenizerFast,␣
 ↪DistilBertForSequenceClassification, Trainer, TrainingArguments
from datasets import Dataset
import pandas as pd
import numpy as np
from sklearn.metrics import classification_report, roc_auc_score

# Load training and testing datasets
df = pd.read_csv("/content/symptom_train.csv")
df_test = pd.read_csv("/content/symptom_test.csv")

# Load the DistilBERT tokenizer
tokenizer = DistilBertTokenizerFast.from_pretrained("distilbert-base-uncased")

# Data Preprocessing: prepare Huggingface Dataset objects
# Rename columns to 'text' and 'label' for Huggingface compatibility
train_dataset = Dataset.from_pandas(df[["Abstract", "MentionsSymptom"]].
 ↪rename(columns={"Abstract": "text", "MentionsSymptom": "label"}))
test_dataset = Dataset.from_pandas(df_test[["Abstract", "MentionsSymptom"]].
 ↪rename(columns={"Abstract": "text", "MentionsSymptom": "label"}))

# Tokenize the datasets
train_dataset = train_dataset.map(lambda x: tokenizer(x["text"],␣
 ↪truncation=True, padding="max_length"), batched=True)
test_dataset = test_dataset.map(lambda x: tokenizer(x["text"], truncation=True,␣
 ↪padding="max_length"), batched=True)

# Define label map and load model
id2label = {0: "No Symptom", 1: "Mentions Symptom"}
label2id = {"No Symptom": 0, "Mentions Symptom": 1}

# Load DistilBERT model for sequence classification
from transformers import AutoModelForSequenceClassification

model = AutoModelForSequenceClassification.from_pretrained(
    "distilbert-base-uncased",
    num_labels=2, # Binary classification (0 or 1)
    id2label=id2label,
    label2id=label2id
)


# Set training arguments
from transformers import TrainingArguments
from transformers import (AutoTokenizer, DataCollatorWithPadding,␣
 ↪AutoModelForSequenceClassification,
    TrainingArguments, Trainer)

training_args = TrainingArguments(
    output_dir="text_classification_model",
    learning_rate=2e-5,
    per_device_train_batch_size=16,
    per_device_eval_batch_size=16,
    num_train_epochs=4,
    weight_decay=0.01,
    eval_strategy="epoch",                      # run eval at the end of each␣
 ↪epoch
    save_strategy="epoch",
    load_best_model_at_end=True,
    push_to_hub=False,
    report_to="none",
    fp16=True  # # Use FP16 (faster on GPUs) with Colab
)


# Define custom evaluation metrics
def compute_metrics(pred):
    labels = pred.label_ids
    preds = np.argmax(pred.predictions, axis=1)
    auc = roc_auc_score(labels, pred.predictions[:, 1])
    report = classification_report(labels, preds, output_dict=True)
    return {
        "accuracy": report["accuracy"],        # Overall accuracy
        "precision": report["1"]["precision"], # Precision for class '1'␣
 ↪(Mentions Symptom)
        "recall": report["1"]["recall"],       # Recall for class '1'
        "f1": report["1"]["f1-score"],         # F1-score for class '1'
        "roc_auc": auc                         # ROC-AUC score
    }


# Initialize Trainer object
trainer = Trainer(
    model=model,
    args=training_args,
    train_dataset=train_dataset,
    eval_dataset=test_dataset,
    compute_metrics=compute_metrics
)
```

```python
trainer.train() # Start training
trainer.evaluate() # Final evaluation on the test set
```

```
/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(
```

```
tokenizer_config.json:   0%|          | 0.00/48.0 [00:00<?, ?B/s]
```

```
vocab.txt:   0%|          | 0.00/232k [00:00<?, ?B/s]
```

```
tokenizer.json:   0%|          | 0.00/466k [00:00<?, ?B/s]
```

```
config.json:   0%|          | 0.00/483 [00:00<?, ?B/s]
```

```
Map:   0%|          | 0/6727 [00:00<?, ? examples/s]
```

```
Map:   0%|          | 0/1442 [00:00<?, ? examples/s]
```

```
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.safetensors:   0%|          | 0.00/268M [00:00<?, ?B/s]
```

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

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

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

```
[ ]: {'eval_loss': 0.06803501397371292,
      'eval_accuracy': 0.986130374479889,
      'eval_precision': 0.9933110367892977,
      'eval_recall': 0.9428571428571428,
      'eval_f1': 0.9674267100977199,
      'eval_roc_auc': 0.9878720018027916,
      'eval_runtime': 5.8504,
      'eval_samples_per_second': 246.48,
      'eval_steps_per_second': 15.555,
      'epoch': 4.0}
```

[ ]:

```
trainer.save_model("./text_classification_model")
tokenizer.save_pretrained("./text_classification_model")  # store tokenizer
```

[ ]:

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

[ ]:

```
import os
print(os.listdir("./text_classification_model"))
```

```
['checkpoint-1684', 'model.safetensors', 'vocab.txt', 'checkpoint-842',
'tokenizer_config.json', 'special_tokens_map.json', 'checkpoint-1263',
'config.json', 'training_args.bin', 'checkpoint-421', 'tokenizer.json']
```

[ ]:

```
import pandas as pd
import torch
from transformers import DistilBertTokenizerFast,␣
 ↪DistilBertForSequenceClassification

# Load the symptom_test.csv dataset
df = pd.read_csv("/content/symptom_test.csv")
# Randomly sample 20 abstracts from the dataset
# Setting random_state ensures reproducibility
sample_df = df.sample(n=20, random_state=42)
sample_texts = sample_df["Abstract"].tolist()
sample_labels = sample_df["MentionsSymptom"].tolist()

# Load tokenizer and model from the saved checkpoint folder
tokenizer = DistilBertTokenizerFast.from_pretrained("./
 ↪text_classification_model")
model = DistilBertForSequenceClassification.from_pretrained("./
 ↪text_classification_model")

model.eval()

# Apply tokenizer with padding and truncation, return PyTorch tensors
inputs = tokenizer(sample_texts, padding=True, truncation=True,␣
 ↪return_tensors="pt")

# Disable gradient computation for faster inference and reduced memory usage
```

```python
with torch.no_grad():
    outputs = model(**inputs)
    logits = outputs.logits
    preds = torch.argmax(logits, dim=1).cpu().numpy()

# Create a DataFrame combining abstracts, true labels, and DistilBERT predictions
results_df = pd.DataFrame({
    "Abstract Snippet": sample_texts,
    "True Label (MentionsSymptom)": sample_labels,
    "DistilBERT Prediction": preds
})

# Set pandas option to display long text fields without truncation
pd.set_option('display.max_colwidth', 200)
print(results_df)
```

```
                           Abstract Snippet  \
0   Amyloid precursor protein (APP) plays a central role in the pathophysiology
of Alzheimer's disease (AD). The accumulation of beta-amyloid protein is
believed to be a crucial step in the developmen…
1   Aging and Alzheimer's disease (AD) exhibit sex differences in several
biological processes, including demyelination. In a recent study, Lopez-Lee et
al. uncover the contributions of sex chromosome…
2   Shift work, the proven circadian rhythm-disrupting behavior, has been linked
to the increased risk of Alzheimer's disease (AD). However, the putative causal
effect and potential mechanisms of shif…
3   The gut-brain axis has emerged as a key player in the regulation of brain
function and cognitive health. Gut microbiota dysbiosis has been observed in
preclinical models of Alzheimer's disease and…
4   Neurodegenerative disorders are characterized by complex neurobiological
changes that are reflected in biomarker alterations detectable in blood,
cerebrospinal fluid (CSF) and with brain imaging. …
5   Neurofilament light chain (NfL) is a promising biomarker for
neurodegenerative diseases, measurable in both CSF and blood upon neuroaxonal
damage. While CSF analysis was traditionally used, blood-…
6   The integration of Electroencephalogram (EEG) measurements with machine
learning holds the promise of enhancing diagnostic accuracy and providing
personalized insights into the progression of neur…
7   Alzheimer's disease (AD) is characterized by progressive cognitive decline
and synaptic dysfunction, largely driven by amyloid plaques and neurofibrillary
tangles (NFTs) composed of hyperphosphory…
8   Diagnostic performance of optical coherence tomography (OCT) to detect
Alzheimer's disease (AD) and mild cognitive impairment (MCI) remains limited. We
aimed to develop a deep-learning algorithm u…
9   Falls guidelines recommendations for individuals classified as 'not-at-risk'
range from no further actions to offering education and exercises. However,
there is a scarcity of prospective studies …
```

```
10  Exercise improves cognitive function in Alzheimer's disease (AD) via
mechanism that are not fully clear. Here, we first examined the effect of
voluntary exercise training (VET) on energy metabolis…
11  Alzheimer's disease (AD) and Parkinson's disease (PD) are multifactorial,
chronic diseases involving neurodegeneration. According to recent studies, it is
hypothesized that the intraneuronal and p…
12  As artificial intelligence evolves, integrating speech processing into home
healthcare (HHC) workflows is increasingly feasible. Audio-recorded
communications enhance risk identification models, w…
13  Curcumin has been proposed as a potential treatment for Alzheimer's disease
(AD) due to its ability to inhibit amyloid-  (A ) peptide aggregates and to
destabilise pre-formed ones. Derivative 27 w…
14  Previous evidence suggests that infectious diseases may contribute to the
development of neurodegenerative diseases (NDDs) while individuals with
hyperglycemia may be at increased risk for both in…
15  Amnestic mild cognitive impairment (aMCI) is considered as an intermediate
stage of Alzheimer's disease, but no MRI biomarkers currently distinguish aMCI
from healthy individuals effectively. Frac…
16  Progress in understanding the causes of physiological and behavioral changes
in post-menopausal women is hampered by the paucity of animal models that
accurately recapitulate these age-associated …
17  The deposition of amyloid-  (A ) aggregates and metal ions within senile
plaques is a hallmark of Alzheimer's disease (AD). Among the modifications
observed in A  peptides, <i>N</i>-terminal trunc…
18  Microelectrode arrays (MEAs) permit recordings with high electrode counts,
thus generating complex datasets that would benefit from precise neuronal spike
sorting for meaningful data extraction. N…
19  Alzheimer's disease (AD) is a debilitating neurodegenerative disease that is
marked by profound neurovascular dysfunction and significant cell-specific
alterations in the brain vasculature. Recent…

    True Label (MentionsSymptom)  DistilBERT Prediction
0                              0                      0
1                              0                      0
2                              0                      0
3                              0                      0
4                              0                      0
5                              0                      0
6                              0                      0
7                              1                      1
8                              1                      1
9                              0                      0
10                             0                      0
11                             0                      0
12                             0                      0
13                             0                      0
14                             0                      0
15                             1                      1
```

16	1	1
17	0	0
18	0	0
19	0	0