Text Classification Using Transformer Networks (BERT)

Sofía Cantú Talamantes A01571120

Some initialization:

In [1]:

```
import random
import torch
import numpy as np
import pandas as pd
from tqdm.notebook import tqdm

# enable tqdm in pandas
tqdm.pandas()

# set to True to use the gpu (if there is one available)
use_gpu = True

# select device
device = torch.device('cuda' if use_gpu and torch.cuda.is_available() else '
print(f'device: {device.type}')

# random seed
seed = 1122

# set random seed
if seed is not None:
    print(f'random seed: {seed}')
    random.seed(seed)
    np.random.seed(seed)
    torch.manual_seed(seed)
```

```
device: cuda
random seed: 1122
```

Read the train/dev/test datasets and create a HuggingFace `Dataset` object:

In [2]:

```
def read_data(filename):
    # read csv file
    df = pd.read_csv(filename, header=None)
    # add column names
    df.columns = ['label', 'title', 'description']
    # make labels zero-based
    df['label'] -= 1
    # concatenate title and description, and remove backslashes
    df['text'] = df['title'] + " " + df['description']
    df['text'] = df['text'].str.replace('\\', ' ', regex=False)
    return df
```

In [3]:

```
#labels = open('data/ag_news_csv/classes.txt').read().splitlines()
#train_df = read_data('data/ag_news_csv/train.csv')
#test_df = read_data('data/ag_news_csv/test.csv')
labels = open('/kaggle/input/ag-news-dataset/data/ag_news_csv/classes.txt').
train_df = read_data('/kaggle/input/ag-news-dataset/data/ag_news_csv/train.c
test_df = read_data('/kaggle/input/ag-news-dataset/data/ag_news_csv/test.csv
train_df
```

Out[3]:

	label	title	description	text
0	2	Wall St. Bears Claw Back Into the Black (Reuters)	Reuters - Short-sellers, Wall Street's dwindli...	Wall St. Bears Claw Back Into the Black (Reute...
1	2	Carlyle Looks Toward Commercial Aerospace (Reu...	Reuters - Private investment firm Carlyle Grou...	Carlyle Looks Toward Commercial Aerospace (Reu...
2	2	Oil and Economy Cloud Stocks' Outlook (Reuters)	Reuters - Soaring crude prices plus worries\ab...	Oil and Economy Cloud Stocks' Outlook (Reuters...
3	2	Iraq Halts Oil Exports from Main Southern Pipe...	Reuters - Authorities have halted oil export\f...	Iraq Halts Oil Exports from Main Southern Pipe...
4	2	Oil prices soar to all-time record, posing new...	AFP - Tearaway world oil prices, toppling reco...	Oil prices soar to all-time record, posing new...
...	...	...	...	...
119995	0	Pakistan's Musharraf Says Won't Quit as Army C...	KARACHI (Reuters) - Pakistani President Perve...	Pakistan's Musharraf Says Won't Quit as Army C...
119996	1	Renteria signing a top-shelf deal	Red Sox general manager Theo Epstein acknowled...	Renteria signing a top-shelf deal Red Sox gene...
119997	1	Saban not going to Dolphins yet	The Miami Dolphins will put their courtship of...	Saban not going to Dolphins yet The Miami Dolp...
119998	1	Today's NFL games	PITTSBURGH at NY GIANTS Time: 1:30 p.m. Line: ...	Today's NFL games PITTSBURGH at NY GIANTS Time...
119999	1	Nets get Carter from Raptors	INDIANAPOLIS -- All-Star Vince Carter was trad...	Nets get Carter from Raptors INDIANAPOLIS -- A...

120000 rows × 4 columns

In [4]:

```
from sklearn.model_selection import train_test_split

train_df, eval_df = train_test_split(train_df, train_size=0.9)
```

```
train_df.reset_index(inplace=True, drop=True)
eval_df.reset_index(inplace=True, drop=True)

print(f'train rows: {len(train_df.index):,}')
print(f'eval rows: {len(eval_df.index):,}')
print(f'test rows: {len(test_df.index):,}')
```

```
train rows: 108,000
eval rows: 12,000
test rows: 7,600
```

In [5]:

```
from datasets import Dataset, DatasetDict

ds = DatasetDict()
ds['train'] = Dataset.from_pandas(train_df)
ds['validation'] = Dataset.from_pandas(eval_df)
ds['test'] = Dataset.from_pandas(test_df)
ds
```

Out[5]:

```
DatasetDict({
    train: Dataset({
        features: ['label', 'title', 'description', 'text'],
        num_rows: 108000
    })
    validation: Dataset({
        features: ['label', 'title', 'description', 'text'],
        num_rows: 12000
    })
    test: Dataset({
        features: ['label', 'title', 'description', 'text'],
        num_rows: 7600
    })
})
```

Tokenize the texts:

In [6]:

```
!pip install ipywidgets
```

```
Requirement already satisfied: ipywidgets in /opt/conda/lib/python3.10/site-
packages (7.7.1)
Requirement already satisfied: ipykernel>=4.5.1 in /opt/conda/lib/python3.1
0/site-packages (from ipywidgets) (6.29.4)
Requirement already satisfied: ipython-genutils~=0.2.0 in /opt/conda/lib/pyt
hon3.10/site-packages (from ipywidgets) (0.2.0)
Requirement already satisfied: traitlets>=4.3.1 in /opt/conda/lib/python3.1
0/site-packages (from ipywidgets) (5.14.3)
Requirement already satisfied: widgetsnbextension~=3.6.0 in /opt/conda/lib/p
ython3.10/site-packages (from ipywidgets) (3.6.9)
Requirement already satisfied: ipython>=4.0.0 in /opt/conda/lib/python3.10/s
ite-packages (from ipywidgets) (8.21.0)
Requirement already satisfied: jupyterlab-widgets>=1.0.0 in /opt/conda/lib/p
ython3.10/site-packages (from ipywidgets) (3.0.11)
Requirement already satisfied: comm>=0.1.1 in /opt/conda/lib/python3.10/sit
e-packages (from ipykernel>=4.5.1->ipywidgets) (0.2.2)
Requirement already satisfied: debugpy>=1.6.5 in /opt/conda/lib/python3.10/s
ite-packages (from ipykernel>=4.5.1->ipywidgets) (1.8.1)
Requirement already satisfied: jupyter-client>=6.1.12 in /opt/conda/lib/pyth
on3.10/site-packages (from ipykernel>=4.5.1->ipywidgets) (7.4.9)
Requirement already satisfied: jupyter-core!=5.0.*,>=4.12 in /opt/conda/lib/
python3.10/site-packages (from ipykernel>=4.5.1->ipywidgets) (5.7.2)
Requirement already satisfied: matplotlib-inline>=0.1 in /opt/conda/lib/pyth
on3.10/site-packages (from ipykernel>=4.5.1->ipywidgets) (0.1.7)
Requirement already satisfied: nest-asyncio in /opt/conda/lib/python3.10/sit
e-packages (from ipykernel>=4.5.1->ipywidgets) (1.6.0)
Requirement already satisfied: packaging in /opt/conda/lib/python3.10/site-p
ackages (from ipykernel>=4.5.1->ipywidgets) (21.3)
Requirement already satisfied: psutil in /opt/conda/lib/python3.10/site-pack
ages (from ipykernel>=4.5.1->ipywidgets) (5.9.3)
Requirement already satisfied: pyzmq>=24 in /opt/conda/lib/python3.10/site-p
ackages (from ipykernel>=4.5.1->ipywidgets) (26.0.3)
Requirement already satisfied: tornado>=6.1 in /opt/conda/lib/python3.10/sit
e-packages (from ipykernel>=4.5.1->ipywidgets) (6.4.1)
Requirement already satisfied: decorator in /opt/conda/lib/python3.10/site-p
ackages (from ipython>=4.0.0->ipywidgets) (5.1.1)
Requirement already satisfied: jedi>=0.16 in /opt/conda/lib/python3.10/site-
packages (from ipython>=4.0.0->ipywidgets) (0.19.1)
Requirement already satisfied: prompt-toolkit<3.1.0,>=3.0.41 in /opt/conda/l
ib/python3.10/site-packages (from ipython>=4.0.0->ipywidgets) (3.0.47)
Requirement already satisfied: pygments>=2.4.0 in /opt/conda/lib/python3.10/
site-packages (from ipython>=4.0.0->ipywidgets) (2.18.0)
Requirement already satisfied: stack-data in /opt/conda/lib/python3.10/site-
packages (from ipython>=4.0.0->ipywidgets) (0.6.2)
Requirement already satisfied: exceptiongroup in /opt/conda/lib/python3.10/s
ite-packages (from ipython>=4.0.0->ipywidgets) (1.2.0)
Requirement already satisfied: pexpect>4.3 in /opt/conda/lib/python3.10/sit
e-packages (from ipython>=4.0.0->ipywidgets) (4.9.0)
Requirement already satisfied: notebook>=4.4.1 in /opt/conda/lib/python3.10/
site-packages (from widgetsnbextension~=3.6.0->ipywidgets) (6.5.7)
Requirement already satisfied: parso<0.9.0,>=0.8.3 in /opt/conda/lib/python
3.10/site-packages (from jedi>=0.16->ipython>=4.0.0->ipywidgets) (0.8.4)
Requirement already satisfied: entrypoints in /opt/conda/lib/python3.10/sit
e-packages (from jupyter-client>=6.1.12->ipykernel>=4.5.1->ipywidgets) (0.4)
Requirement already satisfied: python-dateutil>=2.8.2 in /opt/conda/lib/pyth
on3.10/site-packages (from jupyter-client>=6.1.12->ipykernel>=4.5.1->ipywidg
ets) (2.9.0.post0)
Requirement already satisfied: platformdirs>=2.5 in /opt/conda/lib/python3.1
0/site-packages (from jupyter-core!=5.0.*,>=4.12->ipykernel>=4.5.1->ipywidge
ts) (3.11.0)
Requirement already satisfied: jinja2 in /opt/conda/lib/python3.10/site-pack
ages (from notebook>=4.4.1->widgetsnbextension~=3.6.0->ipywidgets) (3.1.4)
Requirement already satisfied: argon2-cffi in /opt/conda/lib/python3.10/sit
e-packages (from notebook>=4.4.1->widgetsnbextension~=3.6.0->ipywidgets) (2
3.1.0)
Requirement already satisfied: nbformat in /opt/conda/lib/python3.10/site-pa
ckages (from notebook>=4.4.1->widgetsnbextension~=3.6.0->ipywidgets) (5.1
0.4)
Requirement already satisfied: nbconvert>=5 in /opt/conda/lib/python3.10/sit
e-packages (from notebook>=4.4.1->widgetsnbextension~=3.6.0->ipywidgets)
(6.4.5)
Requirement already satisfied: Send2Trash>=1.8.0 in /opt/conda/lib/python3.1
0/site-packages (from notebook>=4.4.1->widgetsnbextension~=3.6.0->ipywidget
s) (1.8.3)
Requirement already satisfied: terminado>=0.8.3 in /opt/conda/lib/python3.1
0/site-packages (from notebook>=4.4.1->widgetsnbextension~=3.6.0->ipywidget
s) (0.18.1)
Requirement already satisfied: prometheus-client in /opt/conda/lib/python3.1
0/site-packages (from notebook>=4.4.1->widgetsnbextension~=3.6.0->ipywidget
s) (0.20.0)
Requirement already satisfied: nbclassic>=0.4.7 in /opt/conda/lib/python3.1
0/site-packages (from notebook>=4.4.1->widgetsnbextension~=3.6.0->ipywidget
s) (1.1.0)
Requirement already satisfied: ptyprocess>=0.5 in /opt/conda/lib/python3.10/
site-packages (from pexpect>4.3->ipython>=4.0.0->ipywidgets) (0.7.0)
Requirement already satisfied: wcwidth in /opt/conda/lib/python3.10/site-pac
kages (from prompt-toolkit<3.1.0,>=3.0.41->ipython>=4.0.0->ipywidgets)
(0.2.13)
Requirement already satisfied: pyparsing!=3.0.5,>=2.0.2 in /opt/conda/lib/py
thon3.10/site-packages (from packaging->ipykernel>=4.5.1->ipywidgets)
(3.1.2)
Requirement already satisfied: executing>=1.2.0 in /opt/conda/lib/python3.1
0/site-packages (from stack-data->ipython>=4.0.0->ipywidgets) (2.0.1)
Requirement already satisfied: asttokens>=2.1.0 in /opt/conda/lib/python3.1
0/site-packages (from stack-data->ipython>=4.0.0->ipywidgets) (2.4.1)
Requirement already satisfied: pure-eval in /opt/conda/lib/python3.10/site-p
ackages (from stack-data->ipython>=4.0.0->ipywidgets) (0.2.2)
Requirement already satisfied: six>=1.12.0 in /opt/conda/lib/python3.10/sit
e-packages (from asttokens>=2.1.0->stack-data->ipython>=4.0.0->ipywidgets)
(1.16.0)
Requirement already satisfied: notebook-shim>=0.2.3 in /opt/conda/lib/python
3.10/site-packages (from nbclassic>=0.4.7->notebook>=4.4.1->widgetsnbextensi
on~=3.6.0->ipywidgets) (0.2.4)
Requirement already satisfied: mistune<2,>=0.8.1 in /opt/conda/lib/python3.1
0/site-packages (from nbconvert>=5->notebook>=4.4.1->widgetsnbextension~=
3.6.0->ipywidgets) (0.8.4)
Requirement already satisfied: jupyterlab-pygments in /opt/conda/lib/python
3.10/site-packages (from nbconvert>=5->notebook>=4.4.1->widgetsnbextension~=
3.6.0->ipywidgets) (0.3.0)
Requirement already satisfied: bleach in /opt/conda/lib/python3.10/site-pack
ages (from nbconvert>=5->notebook>=4.4.1->widgetsnbextension~=3.6.0->ipywidg
ets) (6.1.0)
```

```
Requirement already satisfied: pandocfilters>=1.4.1 in /opt/conda/lib/python
3.10/site-packages (from nbconvert>=5->notebook>=4.4.1->widgetsnbextension~=
3.6.0->ipywidgets) (1.5.0)
Requirement already satisfied: testpath in /opt/conda/lib/python3.10/site-pa
ckages (from nbconvert>=5->notebook>=4.4.1->widgetsnbextension~=3.6.0->ipywi
dgets) (0.6.0)
Requirement already satisfied: defusedxml in /opt/conda/lib/python3.10/site-
packages (from nbconvert>=5->notebook>=4.4.1->widgetsnbextension~=3.6.0->ipy
widgets) (0.7.1)
Requirement already satisfied: beautifulsoup4 in /opt/conda/lib/python3.10/s
ite-packages (from nbconvert>=5->notebook>=4.4.1->widgetsnbextension~=3.6.0-
>ipywidgets) (4.12.3)
Requirement already satisfied: nbclient<0.6.0,>=0.5.0 in /opt/conda/lib/pyth
on3.10/site-packages (from nbconvert>=5->notebook>=4.4.1->widgetsnbextension
~=3.6.0->ipywidgets) (0.5.13)
Requirement already satisfied: MarkupSafe>=2.0 in /opt/conda/lib/python3.10/
site-packages (from nbconvert>=5->notebook>=4.4.1->widgetsnbextension~=
3.6.0->ipywidgets) (2.1.5)
Requirement already satisfied: fastjsonschema>=2.15 in /opt/conda/lib/python
3.10/site-packages (from nbformat->notebook>=4.4.1->widgetsnbextension~=
3.6.0->ipywidgets) (2.19.1)
Requirement already satisfied: jsonschema>=2.6 in /opt/conda/lib/python3.10/
site-packages (from nbformat->notebook>=4.4.1->widgetsnbextension~=3.6.0->ip
ywidgets) (4.22.0)
Requirement already satisfied: argon2-cffi-bindings in /opt/conda/lib/python
3.10/site-packages (from argon2-cffi->notebook>=4.4.1->widgetsnbextension~=
3.6.0->ipywidgets) (21.2.0)
Requirement already satisfied: attrs>=22.2.0 in /opt/conda/lib/python3.10/si
te-packages (from jsonschema>=2.6->nbformat->notebook>=4.4.1->widgetsnbexten
sion~=3.6.0->ipywidgets) (23.2.0)
Requirement already satisfied: jsonschema-specifications>=2023.03.6 in /opt/
conda/lib/python3.10/site-packages (from jsonschema>=2.6->nbformat->notebook
>=4.4.1->widgetsnbextension~=3.6.0->ipywidgets) (2023.12.1)
Requirement already satisfied: referencing>=0.28.4 in /opt/conda/lib/python
3.10/site-packages (from jsonschema>=2.6->nbformat->notebook>=4.4.1->widgets
nbextension~=3.6.0->ipywidgets) (0.35.1)
Requirement already satisfied: rpds-py>=0.7.1 in /opt/conda/lib/python3.10/s
ite-packages (from jsonschema>=2.6->nbformat->notebook>=4.4.1->widgetsnbexte
nsion~=3.6.0->ipywidgets) (0.18.1)
Requirement already satisfied: jupyter-server<3,>=1.8 in /opt/conda/lib/pyth
on3.10/site-packages (from notebook-shim>=0.2.3->nbclassic>=0.4.7->notebook>
=4.4.1->widgetsnbextension~=3.6.0->ipywidgets) (2.12.5)
Requirement already satisfied: cffi>=1.0.1 in /opt/conda/lib/python3.10/sit
e-packages (from argon2-cffi-bindings->argon2-cffi->notebook>=4.4.1->widgets
nbextension~=3.6.0->ipywidgets) (1.16.0)
Requirement already satisfied: soupsieve>1.2 in /opt/conda/lib/python3.10/si
te-packages (from beautifulsoup4->nbconvert>=5->notebook>=4.4.1->widgetsnbex
tension~=3.6.0->ipywidgets) (2.5)
Requirement already satisfied: webencodings in /opt/conda/lib/python3.10/sit
e-packages (from bleach->nbconvert>=5->notebook>=4.4.1->widgetsnbextension~=
3.6.0->ipywidgets) (0.5.1)
Requirement already satisfied: pycparser in /opt/conda/lib/python3.10/site-p
ackages (from cffi>=1.0.1->argon2-cffi-bindings->argon2-cffi->notebook>=
4.4.1->widgetsnbextension~=3.6.0->ipywidgets) (2.22)
Requirement already satisfied: anyio>=3.1.0 in /opt/conda/lib/python3.10/sit
e-packages (from jupyter-server<3,>=1.8->notebook-shim>=0.2.3->nbclassic>=
0.4.7->notebook>=4.4.1->widgetsnbextension~=3.6.0->ipywidgets) (4.4.0)
Requirement already satisfied: jupyter-events>=0.9.0 in /opt/conda/lib/pytho
n3.10/site-packages (from jupyter-server<3,>=1.8->notebook-shim>=0.2.3->nbcl
assic>=0.4.7->notebook>=4.4.1->widgetsnbextension~=3.6.0->ipywidgets) (0.1
0.0)
Requirement already satisfied: jupyter-server-terminals in /opt/conda/lib/py
thon3.10/site-packages (from jupyter-server<3,>=1.8->notebook-shim>=0.2.3->n
bclassic>=0.4.7->notebook>=4.4.1->widgetsnbextension~=3.6.0->ipywidgets)
(0.5.3)
Requirement already satisfied: overrides in /opt/conda/lib/python3.10/site-p
ackages (from jupyter-server<3,>=1.8->notebook-shim>=0.2.3->nbclassic>=
0.4.7->notebook>=4.4.1->widgetsnbextension~=3.6.0->ipywidgets) (7.7.0)
Requirement already satisfied: websocket-client in /opt/conda/lib/python3.1
0/site-packages (from jupyter-server<3,>=1.8->notebook-shim>=0.2.3->nbclassi
c>=0.4.7->notebook>=4.4.1->widgetsnbextension~=3.6.0->ipywidgets) (1.8.0)
Requirement already satisfied: idna>=2.8 in /opt/conda/lib/python3.10/site-p
ackages (from anyio>=3.1.0->jupyter-server<3,>=1.8->notebook-shim>=0.2.3->nb
classic>=0.4.7->notebook>=4.4.1->widgetsnbextension~=3.6.0->ipywidgets)
(3.7)
Requirement already satisfied: sniffio>=1.1 in /opt/conda/lib/python3.10/sit
e-packages (from anyio>=3.1.0->jupyter-server<3,>=1.8->notebook-shim>=0.2.3-
>nbclassic>=0.4.7->notebook>=4.4.1->widgetsnbextension~=3.6.0->ipywidgets)
(1.3.1)
Requirement already satisfied: typing-extensions>=4.1 in /opt/conda/lib/pyth
on3.10/site-packages (from anyio>=3.1.0->jupyter-server<3,>=1.8->notebook-sh
im>=0.2.3->nbclassic>=0.4.7->notebook>=4.4.1->widgetsnbextension~=3.6.0->ipy
widgets) (4.12.2)
Requirement already satisfied: python-json-logger>=2.0.4 in /opt/conda/lib/p
ython3.10/site-packages (from jupyter-events>=0.9.0->jupyter-server<3,>=1.8-
>notebook-shim>=0.2.3->nbclassic>=0.4.7->notebook>=4.4.1->widgetsnbextension
~=3.6.0->ipywidgets) (2.0.7)
Requirement already satisfied: pyyaml>=5.3 in /opt/conda/lib/python3.10/sit
e-packages (from jupyter-events>=0.9.0->jupyter-server<3,>=1.8->notebook-shi
m>=0.2.3->nbclassic>=0.4.7->notebook>=4.4.1->widgetsnbextension~=3.6.0->ipyw
idgets) (6.0.2)
Requirement already satisfied: rfc3339-validator in /opt/conda/lib/python3.1
0/site-packages (from jupyter-events>=0.9.0->jupyter-server<3,>=1.8->noteboo
k-shim>=0.2.3->nbclassic>=0.4.7->notebook>=4.4.1->widgetsnbextension~=3.6.0-
>ipywidgets) (0.1.4)
Requirement already satisfied: rfc3986-validator>=0.1.1 in /opt/conda/lib/py
thon3.10/site-packages (from jupyter-events>=0.9.0->jupyter-server<3,>=1.8->
notebook-shim>=0.2.3->nbclassic>=0.4.7->notebook>=4.4.1->widgetsnbextension~
=3.6.0->ipywidgets) (0.1.1)
Requirement already satisfied: fqdn in /opt/conda/lib/python3.10/site-packag
es (from jsonschema[format-nongpl]>=4.18.0->jupyter-events>=0.9.0->jupyter-s
erver<3,>=1.8->notebook-shim>=0.2.3->nbclassic>=0.4.7->notebook>=4.4.1->widg
etsnbextension~=3.6.0->ipywidgets) (1.5.1)
Requirement already satisfied: isoduration in /opt/conda/lib/python3.10/sit
e-packages (from jsonschema[format-nongpl]>=4.18.0->jupyter-events>=0.9.0->j
upyter-server<3,>=1.8->notebook-shim>=0.2.3->nbclassic>=0.4.7->notebook>=
4.4.1->widgetsnbextension~=3.6.0->ipywidgets) (20.11.0)
Requirement already satisfied: jsonpointer>1.13 in /opt/conda/lib/python3.1
0/site-packages (from jsonschema[format-nongpl]>=4.18.0->jupyter-events>=
0.9.0->jupyter-server<3,>=1.8->notebook-shim>=0.2.3->nbclassic>=0.4.7->noteb
ook>=4.4.1->widgetsnbextension~=3.6.0->ipywidgets) (2.4)
Requirement already satisfied: uri-template in /opt/conda/lib/python3.10/sit
```

```
e-packages (from jsonschema[format-nongpl]>=4.18.0->jupyter-events>=0.9.0->j
upyter-server<3,>=1.8->notebook-shim>=0.2.3->nbclassic>=0.4.7->notebook>=
4.4.1->widgetsnbextension~=3.6.0->ipywidgets) (1.3.0)
Requirement already satisfied: webcolors>=1.11 in /opt/conda/lib/python3.10/
site-packages (from jsonschema[format-nongpl]>=4.18.0->jupyter-events>=
0.9.0->jupyter-server<3,>=1.8->notebook-shim>=0.2.3->nbclassic>=0.4.7->noteb
ook>=4.4.1->widgetsnbextension~=3.6.0->ipywidgets) (24.6.0)
Requirement already satisfied: arrow>=0.15.0 in /opt/conda/lib/python3.10/si
te-packages (from isoduration->jsonschema[format-nongpl]>=4.18.0->jupyter-ev
ents>=0.9.0->jupyter-server<3,>=1.8->notebook-shim>=0.2.3->nbclassic>=0.4.7-
>notebook>=4.4.1->widgetsnbextension~=3.6.0->ipywidgets) (1.3.0)
Requirement already satisfied: types-python-dateutil>=2.8.10 in /opt/conda/l
ib/python3.10/site-packages (from arrow>=0.15.0->isoduration->jsonschema[for
mat-nongpl]>=4.18.0->jupyter-events>=0.9.0->jupyter-server<3,>=1.8->noteboo
k-shim>=0.2.3->nbclassic>=0.4.7->notebook>=4.4.1->widgetsnbextension~=3.6.0-
>ipywidgets) (2.9.0.20240316)
```

In [7]:

```python
from transformers import AutoTokenizer

transformer_name = 'bert-base-cased'
#tokenizer = AutoTokenizer.from_pretrained(transformer_name)
tokenizer = AutoTokenizer.from_pretrained(
    transformer_name, clean_up_tokenization_spaces=True, quiet=True
)
```

```
tokenizer_config.json:   0%|          | 0.00/49.0 [00:00<?, ?B/s]
config.json:   0%|          | 0.00/570 [00:00<?, ?B/s]
vocab.txt:   0%|          | 0.00/213k [00:00<?, ?B/s]
tokenizer.json:   0%|          | 0.00/436k [00:00<?, ?B/s]
```

In [9]:

```python
import logging
logging.disable(logging.WARNING)  # Suppress progress bar and warnings

def tokenize(examples):
    return tokenizer(examples['text'], truncation=True)

train_ds = ds['train'].map(
    tokenize,
    batched=True,
    remove_columns=['title', 'description', 'text']
)
eval_ds = ds['validation'].map(
    tokenize,
    batched=True,
    remove_columns=['title', 'description', 'text']
)

train_ds.to_pandas()
```

```
Map:   0%|          | 0/108000 [00:00<?, ? examples/s]
Map:   0%|          | 0/12000 [00:00<?, ? examples/s]
```

Out[9]:

	label	input_ids	token_type_ids	attention_mask
0	2	[101, 16752, 13335, 1186, 2101, 6690, 9717, 11...	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
1	1	[101, 145, 11680, 17308, 9741, 2428, 150, 1469...	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
2	2	[101, 1418, 14099, 27086, 1494, 1114, 4031, 11...	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
3	1	[101, 2404, 117, 6734, 1996, 118, 1565, 5465, ...	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
4	3	[101, 142, 10044, 27302, 4317, 1584, 3273, 111...	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
...	...	...	...	...
107995	1	[101, 4922, 2274, 1654, 1112, 10503, 1505, 112...	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
107996	3	[101, 10605, 24632, 11252, 21285, 10221, 118, ...	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
107997	2	[101, 13832, 3484, 11300, 4060, 5058, 112, 188...	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
107998	3	[101, 142, 13675, 3756, 5795, 2445, 1104, 109,...	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
107999	2	[101, 157, 16450, 1658, 5302, 185, 7776, 11006...	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...

108000 rows × 4 columns

Create the transformer model:

In [10]:

```
from torch import nn
from transformers.modeling_outputs import SequenceClassifierOutput
from transformers.models.bert.modeling_bert import BertModel, BertPreTrained

# https://github.com/huggingface/transformers/blob/65659a29cf5a079842e61a63d

class BertForSequenceClassification(BertPreTrainedModel):
    def __init__(self, config):
        super().__init__(config)
        self.num_labels = config.num_labels
        self.bert = BertModel(config)
        self.dropout = nn.Dropout(config.hidden_dropout_prob)
        self.classifier = nn.Linear(config.hidden_size, config.num_labels)
        self.init_weights()

    def forward(self, input_ids=None, attention_mask=None, token_type_ids=No
        outputs = self.bert(
            input_ids,
            attention_mask=attention_mask,
            token_type_ids=token_type_ids,
            **kwargs,
        )
        cls_outputs = outputs.last_hidden_state[:, 0, :]
        cls_outputs = self.dropout(cls_outputs)
        logits = self.classifier(cls_outputs)
        loss = None
        if labels is not None:
            loss_fn = nn.CrossEntropyLoss()
            loss = loss_fn(logits, labels)
        return SequenceClassifierOutput(
            loss=loss,
            logits=logits,
            hidden_states=outputs.hidden_states,
            attentions=outputs.attentions,
        )
```

In [11]:

```
from transformers import AutoConfig

config = AutoConfig.from_pretrained(
    transformer_name,
    num_labels=len(labels),
)

model = (
    BertForSequenceClassification
    .from_pretrained(transformer_name, config=config)
)
```

```
model.safetensors:   0%|          | 0.00/436M [00:00<?, ?B/s]
```

Create the trainer object and train:

In [13]:

```
from transformers import TrainingArguments

num_epochs = 2
batch_size = 24
weight_decay = 0.01
model_name = f'{transformer_name}-sequence-classification'

training_args = TrainingArguments(
    output_dir='./results',           # Directory for model outputs
    save_strategy="no",               # Do not save checkpoints
    num_train_epochs=3,               # Number of epochs
    per_device_train_batch_size=8,   # Batch size for training
    per_device_eval_batch_size=8,    # Batch size for evaluation
    weight_decay=0.01,                # Weight decay
    logging_dir='./logs',            # Directory for logs
)
```

In [14]:

```
from sklearn.metrics import accuracy_score

def compute_metrics(eval_pred):
    y_true = eval_pred.label_ids
    y_pred = np.argmax(eval_pred.predictions, axis=-1)
    return {'accuracy': accuracy_score(y_true, y_pred)}
```

In [16]:

```
from transformers import Trainer
import os

# Ensure wandb is completely disabled
os.environ["WANDB_DISABLED"] = "true"
os.environ["WANDB_MODE"] = "disabled"

# Define training arguments
training_args = TrainingArguments(
    output_dir="./results",
    report_to="none",  # Disable all reporting integrations
    num_train_epochs=3,
    per_device_train_batch_size=16,
    per_device_eval_batch_size=16,
    evaluation_strategy="epoch",
    save_strategy="epoch",
    weight_decay=0.01,
)

# Create Trainer
trainer = Trainer(
    model=model,
    args=training_args,
    compute_metrics=compute_metrics,
    train_dataset=train_ds,
    eval_dataset=eval_ds,
    tokenizer=tokenizer,
)
```

```
/opt/conda/lib/python3.10/site-packages/transformers/training_args.py:1545:
FutureWarning: `evaluation_strategy` is deprecated and will be removed in ve
rsion 4.46 of 🤗 Transformers. Use `eval_strategy` instead
  warnings.warn(
```

In [17]:

```
trainer.train()
```

[20250/20250 1:04:17, Epoch 3/3]

Epoch	Training Loss	Validation Loss	Accuracy
1	0.198000	0.197395	0.938167
2	0.146300	0.201212	0.943667
3	0.070600	0.251510	0.946000

Out[17]:

```
TrainOutput(global_step=20250, training_loss=0.16001928984088662, metrics=
{'train_runtime': 3858.3744, 'train_samples_per_second': 83.973, 'train_ste
ps_per_second': 5.248, 'total_flos': 1.7728815264181632e+16, 'train_loss':
0.16001928984088662, 'epoch': 3.0})
```

Evaluate on the test partition:

In [18]:

```python
test_ds = ds['test'].map(
    tokenize,
    batched=True,
    remove_columns=['title', 'description', 'text'],
)
test_ds.to_pandas()
```

```
Map:   0%|          | 0/7600 [00:00<?, ? examples/s]
```

Out[18]:

	label	input_ids	token_type_ids	attention_mask
0	2	[101, 11284, 1116, 1111, 157, 151, 12966, 1170...	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
1	3	[101, 1109, 6398, 1110, 1212, 131, 2307, 7219,...	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
2	3	[101, 148, 1183, 119, 1881, 16387, 1116, 4468,...	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
3	3	[101, 11689, 15906, 6115, 12056, 1116, 1370, 2...	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
4	3	[101, 11917, 8914, 119, 19294, 4206, 1106, 215...	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
...	...	...	...	...
7595	0	[101, 5596, 1103, 1362, 5284, 5200, 3234, 1384...	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
7596	1	[101, 159, 7874, 1110, 2709, 1114, 13875, 1556...	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
7597	1	[101, 16247, 2972, 9178, 2409, 4271, 140, 1418...	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
7598	2	[101, 126, 1104, 1893, 8167, 10721, 4420, 1107...	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
7599	2	[101, 142, 2064, 4164, 3370, 1154, 13519, 1116...	[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...

7600 rows × 4 columns

In [19]:

```
output = trainer.predict(test_ds)
output
```

Out[19]:

```
PredictionOutput(predictions=array([[ 0.19795685, -4.619936  ,  5.150589  ,
-2.24432   ],
       [-1.4081583 , -2.9251745 , -3.5345733 ,  6.778545  ],
       [-1.3091689 , -3.3134522 , -3.2881854 ,  6.4918365 ],
       ...,
       [-2.041592  ,  8.425348  , -2.4603012 , -3.4557693 ],
       [-0.81975   , -3.7158365 ,  6.2912555 , -3.2601109 ],
       [-2.8590794 , -4.609074  ,  4.9792194 ,  0.66535884]],
      dtype=float32), label_ids=array([2, 3, 3, ..., 1, 2, 2]), metrics={'t
est_loss': 0.2598552703857422, 'test_accuracy': 0.9438157894736842, 'test_r
untime': 24.6787, 'test_samples_per_second': 307.958, 'test_steps_per_secon
d': 19.247})
```

In [20]:

```
from sklearn.metrics import classification_report

y_true = output.label_ids
y_pred = np.argmax(output.predictions, axis=-1)
target_names = labels
print(classification_report(y_true, y_pred, target_names=target_names))
```

```
              precision    recall  f1-score   support

       World       0.96      0.95      0.96      1900
      Sports       0.98      0.99      0.99      1900
    Business       0.92      0.91      0.91      1900
    Sci/Tech       0.91      0.93      0.92      1900

    accuracy                           0.94      7600
   macro avg       0.94      0.94      0.94      7600
weighted avg       0.94      0.94      0.94      7600
```

Descripción de la estructura del pipeline del código del notebook

Inicialización y Configuración:

- Importación de librerías necesarias (torch, transformers, pandas, etc.).
- Configuración del dispositivo de cómputo (CPU o GPU) y la semilla para garantizar reproducibilidad.

Carga y Preprocesamiento de Datos:

- Lectura de los datasets (entrenamiento, validación y prueba) desde archivos CSV.
- Creación de una columna combinada text que concatena el título y descripción de los textos.
- Normalización de datos, como la eliminación de caracteres especiales y ajuste de

las etiquetas (label) para que comiencen desde 0.
- Dividir los datos de entrenamiento en subconjuntos de entrenamiento y validación (90%-10%).

Conversión de Datos a Objetos de HuggingFace:

- Transformación de los DataFrames a objetos del tipo Dataset y DatasetDict para ser utilizados por el modelo.

Tokenización:

- Uso de un tokenizador BERT preentrenado (bert-base-cased) para convertir los textos en secuencias de tokens compatibles con el modelo.
- Aplicación de la tokenización a los datasets mediante mapeo batched y eliminación de columnas innecesarias.

Definición del Modelo:

- Construcción del modelo de clasificación de secuencias (BertForSequenceClassification) basado en BERT, añadiendo una capa lineal para clasificar los datos en categorías específicas.
- Configuración de hiperparámetros del modelo, como el tamaño de la capa oculta y el número de etiquetas.

Entrenamiento del Modelo:

- Configuración de los argumentos de entrenamiento, como el número de épocas (3), el tamaño del batch, la estrategia de evaluación, y el peso de decaimiento.
- Creación de un objeto Trainer de HuggingFace que gestiona el entrenamiento, validación y evaluación.
- Entrenamiento del modelo utilizando los datos tokenizados y supervisando la pérdida y precisión durante las épocas.

Evaluación del Modelo:

- Evaluación del modelo entrenado en el conjunto de prueba mediante métricas de clasificación (precisión, recall, F1-score) y cálculo de la pérdida en prueba.
- Generación de un reporte detallado de clasificación usando classification_report de sklearn.

Predicción y Análisis:

- Uso del modelo entrenado para realizar predicciones en el conjunto de prueba.
- Análisis de las métricas y resultados obtenidos, como precisión global y métricas

específicas para cada clase.