```
!pip install datasets
```

```
!!pip install transformers
```

```
!python -c "from datasets import load_dataset; print(load_dataset('squad', sp
```

```
Downloading: 5.27kB [00:00, 4.87MB/s]
Downloading: 2.36kB [00:00, 2.98MB/s]
Downloading and preparing dataset squad/plain_text (download: 33.51 MiB, gene
  0% 0/2 [00:00<?, ?it/s]
Downloading:   0% 0.00/8.12M [00:00<?, ?B/s]
Downloading:  54% 4.40M/8.12M [00:00<00:00, 43.9MB/s]
Downloading: 10.5MB [00:00, 53.7MB/s]
Downloading: 16.1MB [00:00, 55.1MB/s]
Downloading: 22.0MB [00:00, 56.3MB/s]
Downloading: 30.3MB [00:00, 55.9MB/s]
 50% 1/2 [00:02<00:02,  2.28s/it]
Downloading: 4.85MB [00:00, 52.6MB/s]
100% 2/2 [00:02<00:00,  1.49s/it]
100% 2/2 [00:00<00:00, 1161.70it/s]
Dataset squad downloaded and prepared to /root/.cache/huggingface/datasets/sq
{'id': '5733be284776f41900661182', 'title': 'University_of_Notre_Dame', 'cont
```

```
import datasets
import transformers
```

```
from datasets import load_dataset, load_metric
```

```
dataset = load_dataset("health_fact")
```

```
Using custom data configuration default
Downloading and preparing dataset health_fact/default (download: 23.74 MiB, g
```

Downloading: 100% 24.9M/24.9M [00:01<00:00, 26.8MB/s]

9328/0 [00:01<00:00, 4896.32 examples/s]

886/0 [00:00<00:00, 3363.18 examples/s]

Automatic saving failed. This file was updated remotely or in another tab. Show diff

ingface/datas

100% 3/3 [00:00<00:00, 57.69it/s]

```
1 dataset
```

```
DatasetDict({
    train: Dataset({
```

12m 49s completed at 7:01 PM

```
            features: ['claim_id', 'claim', 'date_published', 'explanation', 'fac
            num_rows: 1225
        })
    })
```

```
1 dataset["train"][0]
```

```
{'claim': '"The money the Clinton Foundation took from from foreign governmen
 'claim_id': '15661',
 'date_published': 'April 26, 2015',
 'explanation': '"Gingrich said the Clinton Foundation ""took money from from
 'fact_checkers': 'Katie Sanders',
 'label': 0,
 'main_text': '"Hillary Clinton is in the political crosshairs as the author
 'sources': 'https://www.wsj.com/articles/clinton-foundation-defends-acceptan
 'subjects': 'Foreign Policy, PunditFact, Newt Gingrich, '}
```

```
1 vals = set()
2 examples = []
3 for elem in dataset["train"]:
4     if elem['label'] not in vals:
5         vals.add(elem['label'])
6         examples.append((elem['label'], elem['claim']))
7
8 for exampl in examples:
9     print(exampl)
```

```
(0, '"The money the Clinton Foundation took from from foreign governments whi
(1, 'Annual Mammograms May Have More False-Positives')
(2, 'Study: Vaccine for Breast, Ovarian Cancer Has Potential')
(3, '"Secretary of State John Kerry ""funneled"" taxpayer money into his daug
(-1, ' A forwarded email that cautions veterans about   questions that maybe
```

```
1 dataset['train'].features.items()
```

```
dict_items([('claim_id', Value(dtype='string', id=None)), ('claim', Value(dty
```

```
1 valid_labels = [0, 1, 2] # 'false', 'mixture', 'true'
```

Automatic saving failed. This file was updated remotely or in another tab. Show diff

```
5 filtered['train'] = dataset["train"].filter(lambda example: example['label'] in
```

```
18 print(len(dataset["validation"]))
```

```
100%                                                  10/10 [00:00<00:00, 28.71ba/s]
train
9513
9832
100%                                                  2/2 [00:00<00:00, 8.86ba/s]
test
1188
1235
100%                                                  2/2 [00:00<00:00, 9.07ba/s]
validation
1173
1225
```

```
1  def concat(example):
2         example['concat'] = example['claim'] + " " + example['main_text']
3         return example
4
5 filtered['train'] = filtered['train'].map(concat)
6 filtered['test'] = filtered['test'].map(concat)
7 filtered['validation'] = filtered['validation'].map(concat)
```

```
100%                                                  9513/9513 [00:02<00:00, 3838.13ex/s]
100%                                                  1188/1188 [00:00<00:00, 3374.46ex/s]
100%                                                  1173/1173 [00:00<00:00, 3583.87ex/s]
```

```
1 filtered
```

```
{'test': Dataset({
     features: ['claim_id', 'claim', 'date_published', 'explanation', 'fact_c
     num_rows: 1188
 }), 'train': Dataset({
     features: ['claim_id', 'claim', 'date_published', 'explanation', 'fact_c
     num_rows: 9513
```

Automatic saving failed. This file was updated remotely or in another tab. Show diff

```
    df = pd.DataFrame(dataset[picks])
    for column, typ in dataset.features.items():
        if isinstance(typ, datasets.ClassLabel):
            df[column] = df[column].transform(lambda i: typ.names[i])
    display(HTML(df.to_html()))
```

```
show_random_elements(filtered["train"],num_examples=3)
```

claim_id	claim	date_published	explanation	fact_checkers	main_t
					"If you're see to tie opponent to controversial what better to do it tha say he cas deciding vo pass them? T exactly wha Nati Republ Senat Committ doing i Colorado Se race betv incum Democrat Mic Bennet and nominee

Automatic saving failed. This file was updated remotely or in another tab. Show diff

passage o
trillion-d
health car
that sla
Medicare, bu

```
from transformers import AutoTokenizer

# tokenizer = AutoTokenizer.from_pretrained("bert-base-cased")
tokenizer = AutoTokenizer.from_pretrained("distilbert-base-uncased")
```

```
Downloading: 100%  28.0/28.0 [00:00<00:00, 647B/s]
Downloading: 100%  483/483 [00:00<00:00, 12.0kB/s]
Downloading: 100%  226k/226k [00:00<00:00, 362kB/s]
Downloading: 100%  455k/455k [00:00<00:00, 648kB/s]
```

```
def tokenize_function(examples):
    return tokenizer(examples["concat"], padding="max_length", truncation=True)


filtered['train'] = filtered['train'].map(tokenize_function, batched=True)
filtered['test'] = filtered['test'].map(tokenize_function, batched=True)
filtered['validation'] = filtered['validation'].map(tokenize_function, batched=
```

Automatic saving failed. This file was updated remotely or in another tab. Show diff

```
import numpy as np
from datasets import load_metric

metric_name = 'accuracy'
metric = load_metric(metric_name)

def compute_metrics(eval_pred):
    logits, labels = eval_pred
    predictions = np.argmax(logits, axis=-1)
    return metric.compute(predictions=predictions, references=labels)
```

Downloading: 3.20k/? [00:00<00:00, 75.4kB/s]

```
from transformers import TrainingArguments


model_name = model_checkpoint.split("/")[-1]
task = "classification"
```

Automatic saving failed. This file was updated remotely or in another tab. Show diff

```
Num examples = 1000
Num Epochs = 5
Instantaneous batch size per device = 16
Total train batch size (w. parallel, distributed & accumulation) = 16
Gradient Accumulation steps = 1
Total optimization steps = 315
```

[315/315 10:53, Epoch 5/5]

Epoch	Training Loss	Validation Loss	Accuracy
1	No log	0.893087	0.613000
2	No log	0.960932	0.606000
3	No log	0.853767	0.637000

Automatic saving failed. This file was updated remotely or in another tab. Show diff

```
  Batch size = 16
```



```
[63/63 00:33]
{'epoch': 5.0,
 'eval_accuracy': 0.642,
 'eval_loss': 0.8556296825408936,
 'eval_runtime': 34.2757,
 'eval_samples_per_second': 29.175,
 'eval_steps_per_second': 1.838}
```

Automatic saving failed. This file was updated remotely or in another tab. Show diff

```
Num examples = 1186
```

Automatic saving failed. This file was updated remotely or in another tab. Show diff

Automatic saving failed. This file was updated remotely or in another tab. Show diff

Automatic saving failed. This file was updated remotely or in another tab. Show diff