```
from setfit import SetFitModel, SetFitTrainer
from sentence_transformers.losses import CosineSimilarityLoss
from datasets import Dataset
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score, f1_score
import pandas as pd
import os
```

```
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages/tqdm/auto.py:21: TqdmWarning: IProgress not
  from .autonotebook import tqdm as notebook_tqdm
```

SetFit MODEL - 2

Model: Roberta Large v1

N = 16

```
# === CONFIG ===

DATA_PATH = "multilang_sarcasm_dataset.csv"
MODEL_PATH = "model/setfit_multilang_sarcasm_roberta_N16"
N_SHOT = 16
MAX_TEST_SAMPLES = 1000

# === LOAD & PREPROCESS ===
df = pd.read_csv(DATA_PATH)

# Filter to English headlines
df = df[df["lang"] == "en"]

# Rename columns to match SETFIT input format
df = df[["article_title", "is_sarcastic"]].rename(columns={"article_title": "text", "is_sarcastic": "label"})
df = df.dropna(subset=["text", "label"])

# === TRAIN/TEST SPLIT ===
train_df, test_df = train_test_split(df, test_size=0.2, stratify=df["label"], random_state=42)

# Few-shot sampling
def sample_few_shot(df, n=32):
    return df.groupby("label").apply(lambda x: x.sample(n=min(n, len(x)), random_state=42)).reset_index(drop=True)

fewshot_train_df = sample_few_shot(train_df, N_SHOT)
test_subset_df = test_df.sample(n=min(len(test_df), MAX_TEST_SAMPLES), random_state=42)

# Convert to HuggingFace datasets
train_dataset = Dataset.from_pandas(fewshot_train_df)
test_dataset = Dataset.from_pandas(test_subset_df)

# === LOAD BASE MODEL (ROBERTA) ===
model = SetFitModel.from_pretrained("sentence-transformers/all-roberta-large-v1")

# === TRAIN SETUP ===
trainer = SetFitTrainer(
    model=model,
    train_dataset=train_dataset,
    eval_dataset=test_dataset,
    loss_class=CosineSimilarityLoss,
    batch_size=16,
    num_iterations=10,
    num_epochs=1,
    column_mapping={"text": "text", "label": "label"},
)

trainer.train()

# Save model
model.save_pretrained(MODEL_PATH)

# Evaluate
y_true = test_dataset["label"]
y_pred = model.predict(test_dataset["text"])
acc = accuracy_score(y_true, y_pred)
f1 = f1_score(y_true, y_pred)
```

```
print(f"SETFIT_RoBERTa_N16 | Accuracy: {acc:.4f} | F1 Score: {f1:.4f}")
```

```
/var/folders/lv/xd91rcv91cq23cjjl0_c93nh0000gn/T/ipykernel_50620/3845588370.py:23: DeprecationWarning: DataFrameGroupBy.appl
  return df.groupby("label").apply(lambda x: x.sample(n=min(n, len(x)), random_state=42)).reset_index(drop=True)
`SentenceTransformer._target_device` has been deprecated, please use `SentenceTransformer.device` instead.
model_head.pkl not found on HuggingFace Hub, initialising classification head with random weights. You should TRAIN this mod
Applying column mapping to training dataset
Generating Training Pairs: 100%|██████████| 10/10 [00:00<00:00, 1994.82it/s]
***** Running training *****
  Num examples = 640
  Num epochs = 1
  Total optimization steps = 40
  Total train batch size = 16
```

Step	Training Loss

```
SETFIT_RoBERTa_N16 | Accuracy: 0.6290 | F1 Score: 0.6618
```

SetFit MODEL - 2

Model: Roberta Large v1

N = 32

```
# === CONFIG ===
DATA_PATH = "multilang_sarcasm_dataset.csv"
MODEL_PATH = "model/setfit_multilang_sarcasm_roberta_N32"
N_SHOT = 32
MAX_TEST_SAMPLES = 1000

# === LOAD & PREPROCESS ===
df = pd.read_csv(DATA_PATH)

# Filter to English headlines
df = df[df["lang"] == "en"]

# Rename columns to match SETFIT input format
df = df[["article_title", "is_sarcastic"]].rename(columns={"article_title": "text", "is_sarcastic": "label"})
df = df.dropna(subset=["text", "label"])

# === TRAIN/TEST SPLIT ===
train_df, test_df = train_test_split(df, test_size=0.2, stratify=df["label"], random_state=42)
```

```
# Few-shot sampling
def sample_few_shot(df, n=32):
    return df.groupby("label").apply(lambda x: x.sample(n=min(n, len(x)), random_state=42)).reset_index(drop=True)

fewshot_train_df = sample_few_shot(train_df, N_SHOT)
test_subset_df = test_df.sample(n=min(len(test_df), MAX_TEST_SAMPLES), random_state=42)

# Convert to HuggingFace datasets
train_dataset = Dataset.from_pandas(fewshot_train_df)
test_dataset = Dataset.from_pandas(test_subset_df)

# === LOAD BASE MODEL (ROBERTA) ===
model = SetFitModel.from_pretrained("sentence-transformers/all-roberta-large-v1")

# === TRAIN SETUP ===
trainer = SetFitTrainer(
    model=model,
    train_dataset=train_dataset,
    eval_dataset=test_dataset,
    loss_class=CosineSimilarityLoss,
    batch_size=16,
    num_iterations=10,
    num_epochs=1,
    column_mapping={"text": "text", "label": "label"},
)

trainer.train()

# Save model
model.save_pretrained(MODEL_PATH)

# Evaluate
y_true = test_dataset["label"]
y_pred = model.predict(test_dataset["text"])
acc = accuracy_score(y_true, y_pred)
f1 = f1_score(y_true, y_pred)

print(f"SETFIT_RoBERTa_N32 | Accuracy: {acc:.4f} | F1 Score: {f1:.4f}")
```

```
/var/folders/lv/xd91rcv91cq23cjjl0_c93nh0000gn/T/ipykernel_50620/1738066910.py:22: DeprecationWarning: DataFrameGroupBy.appl
  return df.groupby("label").apply(lambda x: x.sample(n=min(n, len(x)), random_state=42)).reset_index(drop=True)
`SentenceTransformer._target_device` has been deprecated, please use `SentenceTransformer.device` instead.
model_head.pkl not found on HuggingFace Hub, initialising classification head with random weights. You should TRAIN this mod
Applying column mapping to training dataset
Generating Training Pairs: 100%|██████████| 10/10 [00:00<00:00, 1319.79it/s]
***** Running training *****
  Num examples = 1280
  Num epochs = 1
  Total optimization steps = 80
  Total train batch size = 16
```

[80/80 00:57, Epoch 1/1]

Step	Training Loss

```
SETFIT_RoBERTa_N32 | Accuracy: 0.8000 | F1 Score: 0.7859
```

SetFit MODEL - 2

Model: Roberta Large v1

N = 64

```
# === CONFIG ===

DATA_PATH = "multilang_sarcasm_dataset.csv"
MODEL_PATH = "model/setfit_multilang_sarcasm_roberta_N64"
N_SHOT = 64
MAX_TEST_SAMPLES = 1000

# === LOAD & PREPROCESS ===
df = pd.read_csv(DATA_PATH)

# Filter to English headlines
df = df[df["lang"] == "en"]

# Rename columns to match SETFIT input format
```

```
df = df[["article_title", "is_sarcastic"]].rename(columns={"article_title": "text", "is_sarcastic": "label"})
df = df.dropna(subset=["text", "label"])

# === TRAIN/TEST SPLIT ===
train_df, test_df = train_test_split(df, test_size=0.2, stratify=df["label"], random_state=42)

# Few-shot sampling
def sample_few_shot(df, n=32):
    return df.groupby("label").apply(lambda x: x.sample(n=min(n, len(x)), random_state=42)).reset_index(drop=True)

fewshot_train_df = sample_few_shot(train_df, N_SHOT)
test_subset_df = test_df.sample(n=min(len(test_df), MAX_TEST_SAMPLES), random_state=42)

# Convert to HuggingFace datasets
train_dataset = Dataset.from_pandas(fewshot_train_df)
test_dataset = Dataset.from_pandas(test_subset_df)

# === LOAD BASE MODEL (ROBERTA) ===
model = SetFitModel.from_pretrained("sentence-transformers/all-roberta-large-v1")

# === TRAIN SETUP ===
trainer = SetFitTrainer(
    model=model,
    train_dataset=train_dataset,
    eval_dataset=test_dataset,
    loss_class=CosineSimilarityLoss,
    batch_size=16,
    num_iterations=10,
    num_epochs=1,
    column_mapping={"text": "text", "label": "label"},
)

trainer.train()

# Save model
model.save_pretrained(MODEL_PATH)

# Evaluate
y_true = test_dataset["label"]
y_pred = model.predict(test_dataset["text"])
acc = accuracy_score(y_true, y_pred)
f1 = f1_score(y_true, y_pred)

print(f"SETFIT_RoBERTa_N64 | Accuracy: {acc:.4f} | F1 Score: {f1:.4f}")
```

```
/var/folders/lv/xd91rcv91cq23cjjl0_c93nh0000gn/T/ipykernel_50620/3470306218.py:23: DeprecationWarning: DataFrameGroupBy.appl
  return df.groupby("label").apply(lambda x: x.sample(n=min(n, len(x)), random_state=42)).reset_index(drop=True)
`SentenceTransformer._target_device` has been deprecated, please use `SentenceTransformer.device` instead.
model_head.pkl not found on HuggingFace Hub, initialising classification head with random weights. You should TRAIN this mod
Applying column mapping to training dataset
Generating Training Pairs: 100%|██████████| 10/10 [00:00<00:00, 635.27it/s]
***** Running training *****
  Num examples = 2560
  Num epochs = 1
  Total optimization steps = 160
  Total train batch size = 16
```

Step	Training Loss

```
SETFIT_RoBERTa_N64 | Accuracy: 0.8350 | F1 Score: 0.8269
```