Feature transformation with Amazon SageMaker processing job and Feature Store

Introduction

In this lab you will start with the raw Women's Clothing Reviews dataset and prepare it to train a BERT-based natural language processing (NLP) model. The model will be used to classify customer reviews into positive (1), neutral (0) and negative (-1) sentiment.

You will convert the original review text into machine-readable features used by BERT. To perform the required feature transformation you will configure an Amazon SageMaker processing job, which will be running a custom Python script.

Table of Contents

In [1]:
```
# please ignore warning messages during the installation
!pip install --disable-pip-version-check -q sagemaker==2.35.0
!conda install -q -y pytorch==1.6.0 -c pytorch
!pip install --disable-pip-version-check -q transformers==3.5.1
```

```
WARNING: Running pip as the 'root' user can result in broken permissions
and conflicting behaviour with the system package manager. It is recommen
ded to use a virtual environment instead: https://pip.pypa.io/warnings/ve
nv
Collecting package metadata (current_repodata.json): ...working... done
Solving environment: ...working... done

## Package Plan ##

  environment location: /opt/conda

  added / updated specs:
    - pytorch==1.6.0


The following packages will be UPDATED:

  ca-certificates    conda-forge::ca-certificates-2022.6.1~ --> pkgs/main
::ca-certificates-2022.07.19-h06a4308_0

The following packages will be SUPERSEDED by a higher-priority channel:

  conda              conda-forge::conda-4.14.0-py37h89c186~ --> pkgs/main
::conda-4.14.0-py37h06a4308_0


Preparing transaction: ...working... done
Verifying transaction: ...working... done
Executing transaction: ...working... done
Retrieving notices: ...working... done
WARNING: Running pip as the 'root' user can result in broken permissions
and conflicting behaviour with the system package manager. It is recommen
ded to use a virtual environment instead: https://pip.pypa.io/warnings/ve
nv
```

In [2]:

```
import boto3
import sagemaker
import botocore

config = botocore.config.Config(user_agent_extra='dlai-pds/c2/w1')

# low-level service client of the boto3 session
sm = boto3.client(service_name='sagemaker',
                  config=config)

featurestore_runtime = boto3.client(service_name='sagemaker-featurestore-
                                    config=config)

sess = sagemaker.Session(sagemaker_client=sm,
                         sagemaker_featurestore_runtime_client=featuresto

bucket = sess.default_bucket()
role = sagemaker.get_execution_role()
region = sess.boto_region_name
```

1. Configure the SageMaker Feature Store

1.1. Configure dataset

The raw dataset is in the public S3 bucket. Let's start by specifying the S3 location of it:

In [3]:
```
raw_input_data_s3_uri = 's3://dlai-practical-data-science/data/raw/'
print(raw_input_data_s3_uri)
```

```
s3://dlai-practical-data-science/data/raw/
```

List the files in the S3 bucket (in this case it will be just one file):

In [4]:
```
!aws s3 ls $raw_input_data_s3_uri
```

```
2021-04-30 02:21:06    8457214 womens_clothing_ecommerce_reviews.csv
```

1.2. Configure the SageMaker feature store

As the result of the transformation, in addition to generating files in S3 bucket, you will also save the transformed data in the **Amazon SageMaker Feature Store** to be used by others in your organization, for example.

To configure a Feature Store you need to setup a **Feature Group**. This is the main resource containing all of the metadata related to the data stored in the Feature Store. A Feature Group should contain a list of **Feature Definitions**. A Feature Definition consists of a name and the data type. The Feature Group also contains an online store configuration and an offline store configuration controlling where the data is stored. Enabling the online store allows quick access to the latest value for a record via the GetRecord API. The offline store allows storage of the data in your S3 bucket. You will be using the offline store in this lab.

Let's setup the Feature Group name and the Feature Store offline prefix in S3 bucket (you will use those later in the lab):

In [5]:
```
import time
timestamp = int(time.time())

feature_group_name = 'reviews-feature-group-' + str(timestamp)
feature_store_offline_prefix = 'reviews-feature-store-' + str(timestamp)

print('Feature group name: {}'.format(feature_group_name))
print('Feature store offline prefix in S3: {}'.format(feature_store_offli
```

```
Feature group name: reviews-feature-group-1661803330
Feature store offline prefix in S3: reviews-feature-store-1661803330
```

Taking two features from the original raw dataset (`Review Text` and `Rating`), you will transform it preparing to be used for the model training and then to be saved in the Feature Store. Here you will define the related features to be stored as a list of `FeatureDefinition` .

In [6]:
```
from sagemaker.feature_store.feature_definition import (
    FeatureDefinition,
    FeatureTypeEnum,
)

feature_definitions= [
    # unique ID of the review
    FeatureDefinition(feature_name='review_id', feature_type=FeatureTypeE
    # ingestion timestamp
    FeatureDefinition(feature_name='date', feature_type=FeatureTypeEnum.S
    # sentiment: -1 (negative), 0 (neutral) or 1 (positive). It will be f
    FeatureDefinition(feature_name='sentiment', feature_type=FeatureTypeE
    # label ID of the target class (sentiment)
    FeatureDefinition(feature_name='label_id', feature_type=FeatureTypeEn
    # reviews encoded with the BERT tokenizer
    FeatureDefinition(feature_name='input_ids', feature_type=FeatureTypeE
    # original Review Text
    FeatureDefinition(feature_name='review_body', feature_type=FeatureTyp
    # train/validation/test label
    FeatureDefinition(feature_name='split_type', feature_type=FeatureType
]
```

Exercise 1

Create the feature group using the feature definitions defined above.

Instructions: Use the `FeatureGroup` function passing the defined above feature group name and the feature definitions.

```
feature_group = FeatureGroup(
    name=..., # Feature Group name
    feature_definitions=..., # a list of Feature Definitions
    sagemaker_session=sess # SageMaker session
)
```

In [7]:
```
from sagemaker.feature_store.feature_group import FeatureGroup

feature_group = FeatureGroup(
    ### BEGIN SOLUTION - DO NOT delete this comment for grading purposes
    name=feature_group_name, # Replace None
    feature_definitions=feature_definitions, # Replace None
    ### END SOLUTION - DO NOT delete this comment for grading purposes
    sagemaker_session=sess
)

print(feature_group)
```

```
FeatureGroup(name='reviews-feature-group-1661803330', sagemaker_session=<
sagemaker.session.Session object at 0x7fd212934e10>, feature_definitions=
[FeatureDefinition(feature_name='review_id', feature_type=<FeatureTypeEnu
m.STRING: 'String'>), FeatureDefinition(feature_name='date', feature_type
=<FeatureTypeEnum.STRING: 'String'>), FeatureDefinition(feature_name='sen
timent', feature_type=<FeatureTypeEnum.STRING: 'String'>), FeatureDefinit
ion(feature_name='label_id', feature_type=<FeatureTypeEnum.STRING: 'Strin
g'>), FeatureDefinition(feature_name='input_ids', feature_type=<FeatureTy
peEnum.STRING: 'String'>), FeatureDefinition(feature_name='review_body',
feature_type=<FeatureTypeEnum.STRING: 'String'>), FeatureDefinition(featu
re_name='split_type', feature_type=<FeatureTypeEnum.STRING: 'String'>)])
```

You will use the defined Feature Group later in this lab, the actual creation of the Feature Group will take place in the processing job. Now let's move into the setup of the processing job to transform the dataset.

2. Transform the dataset

You will configure a SageMaker processing job to run a custom Python script to balance and transform the raw data into a format used by BERT model.

Set the transformation parameters including the instance type, instance count, and train/validation/test split percentages. For the purposes of this lab, you will use a relatively small instance type. Please refer to this link for additional instance types that may work for your use case outside of this lab.

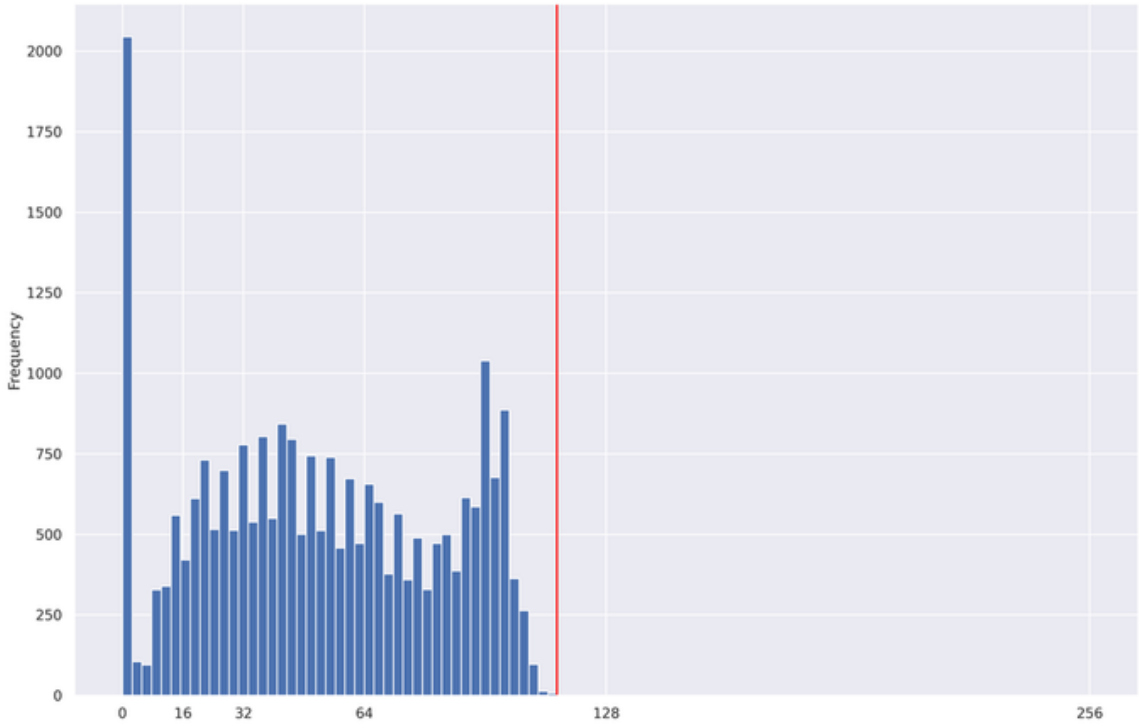
You can also choose whether you want to balance the dataset or not. In this case, you will balance the dataset to avoid class imbalance in the target variable, `sentiment`.

Another important parameter of the model is the `max_seq_length`, which specifies the maximum length of the classified reviews for the RoBERTa model. If the sentence is shorter than the maximum length parameter, it will be padded. In another case, when the sentence is longer, it will be truncated from the right side.

Since a smaller `max_seq_length` leads to faster training and lower resource utilization, you want to find the smallest power-of-2 that captures `100%` of our reviews. For this dataset, the `100th` percentile is `115`. However, it's best to stick with powers-of-2 when using BERT. So let's choose `128` as this is the smallest power-of-2 greater than `115`. You will see below how the shorter sentences will be padded to a maximum length.

```
mean        52.512374
std         31.387048
min          1.000000
10%         10.000000
20%         22.000000
30%         32.000000
```

```
40%          41.000000
50%          51.000000
60%          61.000000
70%          73.000000
80%          88.000000
90%          97.000000
100%        115.000000
max         115.000000
```



In [8]:
```
processing_instance_type='ml.c5.xlarge'
processing_instance_count=1
train_split_percentage=0.90
validation_split_percentage=0.05
test_split_percentage=0.05
balance_dataset=True
max_seq_length=128
```

To balance and transform our data, you will use a scikit-learn-based processing job. This is essentially a generic Python processing job with scikit-learn pre-installed. You can specify the version of scikit-learn you wish to use. Also pass the SageMaker execution role, processing instance type and instance count.

In [9]:
```
from sagemaker.sklearn.processing import SKLearnProcessor

processor = SKLearnProcessor(
    framework_version='0.23-1',
    role=role,
    instance_type=processing_instance_type,
    instance_count=processing_instance_count,
    env={'AWS_DEFAULT_REGION': region},
    max_runtime_in_seconds=7200
)
```

The processing job will be running the Python code from the file `src/prepare_data.py`. In the following exercise you will review the contents of the file and familiarize yourself with main parts of it.

Exercise 2

1. Open the file src/prepare_data.py. Go through the comments to understand its content.
2. Find and review the `convert_to_bert_input_ids()` function, which contains the RoBERTa `tokenizer` configuration.
3. Complete method `encode_plus` of the RoBERTa `tokenizer`. Pass the `max_seq_length` as a value for the argument `max_length`. It defines a pad to a maximum length specified.
4. Save the file src/prepare_data.py (with the menu command File -> Save Python File).

This cell will take approximately 1-2 minutes to run.

In [10]:
```
import sys, importlib
sys.path.append('src/')

# import the `prepare_data.py` module
import prepare_data

# reload the module if it has been previously loaded
if 'prepare_data' in sys.modules:
    importlib.reload(prepare_data)

input_ids = prepare_data.convert_to_bert_input_ids("this product is great

updated_correctly = False

if len(input_ids) != max_seq_length:
    print('#################################################################
    print('Please check that the function \'convert_to_bert_input_ids\' i
    print('#################################################################
    raise Exception('Please check that the function \'convert_to_bert_inp
else:
    print('##################')
    print('Updated correctly!')
    print('##################')

    updated_correctly = True
```

```
##################
Updated correctly!
##################
```

Review the results of tokenization for the given example (|*"this product is great!"*|):

In [11]:
```
input_ids = prepare_data.convert_to_bert_input_ids("this product is great

print(input_ids)
print('Length of the sequence: {}'.format(len(input_ids)))
```

```
[0, 9226, 1152, 16, 372, 328, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1
, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1
, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1
, 1, 1, 1, 1, 1, 1, 1, 1, 1]
Length of the sequence: 128
```

Launch the processing job with the custom script passing defined above parameters.

In [12]:
```
from sagemaker.processing import ProcessingInput, ProcessingOutput

if (updated_correctly):

    processor.run(code='src/prepare_data.py',
                  inputs=[
                        ProcessingInput(source=raw_input_data_s3_uri,
                                        destination='/opt/ml/processing/input
                                        s3_data_distribution_type='ShardedByS
                  ],
                  outputs=[
                        ProcessingOutput(output_name='sentiment-train',
                                         source='/opt/ml/processing/output/se
                                         s3_upload_mode='EndOfJob'),
                        ProcessingOutput(output_name='sentiment-validation',
                                         source='/opt/ml/processing/output/se
                                         s3_upload_mode='EndOfJob'),
                        ProcessingOutput(output_name='sentiment-test',
                                         source='/opt/ml/processing/output/se
                                         s3_upload_mode='EndOfJob')
                  ],
                  arguments=['--train-split-percentage', str(train_split_perc
                             '--validation-split-percentage', str(validation_
                             '--test-split-percentage', str(test_split_percen
                             '--balance-dataset', str(balance_dataset),
                             '--max-seq-length', str(max_seq_length),
                             '--feature-store-offline-prefix', str(feature_st
                             '--feature-group-name', str(feature_group_name)
                  ],
                  logs=True,
                  wait=False)

else:
    print('########################################')
    print('Please update the code correctly above.')
    print('########################################')
```

```
Job Name:  sagemaker-scikit-learn-2022-08-29-20-16-18-304
Inputs:  [{'InputName': 'input-1', 'AppManaged': False, 'S3Input': {'S3Ur
i': 's3://dlai-practical-data-science/data/raw/', 'LocalPath': '/opt/ml/p
rocessing/input/data/', 'S3DataType': 'S3Prefix', 'S3InputMode': 'File',
'S3DataDistributionType': 'ShardedByS3Key', 'S3CompressionType': 'None'}}
, {'InputName': 'code', 'AppManaged': False, 'S3Input': {'S3Uri': 's3://s
agemaker-us-east-1-504458356961/sagemaker-scikit-learn-2022-08-29-20-16-1
8-304/input/code/prepare_data.py', 'LocalPath': '/opt/ml/processing/input
/code', 'S3DataType': 'S3Prefix', 'S3InputMode': 'File', 'S3DataDistribut
ionType': 'FullyReplicated', 'S3CompressionType': 'None'}}]
Outputs:  [{'OutputName': 'sentiment-train', 'AppManaged': False, 'S3Outp
ut': {'S3Uri': 's3://sagemaker-us-east-1-504458356961/sagemaker-scikit-le
arn-2022-08-29-20-16-18-304/output/sentiment-train', 'LocalPath': '/opt/m
l/processing/output/sentiment/train', 'S3UploadMode': 'EndOfJob'}}, {'Out
putName': 'sentiment-validation', 'AppManaged': False, 'S3Output': {'S3Ur
i': 's3://sagemaker-us-east-1-504458356961/sagemaker-scikit-learn-2022-08
-29-20-16-18-304/output/sentiment-validation', 'LocalPath': '/opt/ml/proc
essing/output/sentiment/validation', 'S3UploadMode': 'EndOfJob'}}, {'Outp
utName': 'sentiment-test', 'AppManaged': False, 'S3Output': {'S3Uri': 's3
://sagemaker-us-east-1-504458356961/sagemaker-scikit-learn-2022-08-29-20-
16-18-304/output/sentiment-test', 'LocalPath': '/opt/ml/processing/output
/sentiment/test', 'S3UploadMode': 'EndOfJob'}}]
```

You can see the information about the processing jobs using the `describe` function. The result is in dictionary format. Let's pull the processing job name:

In [13]:
```
scikit_processing_job_name = processor.jobs[-1].describe()['ProcessingJob

print('Processing job name: {}'.format(scikit_processing_job_name))
```

```
Processing job name: sagemaker-scikit-learn-2022-08-29-20-16-18-304
```

Exercise 3

Pull the processing job status from the processing job description.

Instructions: Print the keys of the processing job description dictionary, choose the one related to the status of the processing job and print the value of it.

In [14]:
```
print(processor.jobs[-1].describe().keys())
```

```
dict_keys(['ProcessingInputs', 'ProcessingOutputConfig', 'ProcessingJobNa
me', 'ProcessingResources', 'StoppingCondition', 'AppSpecification', 'Env
ironment', 'RoleArn', 'ProcessingJobArn', 'ProcessingJobStatus', 'LastMod
ifiedTime', 'CreationTime', 'ResponseMetadata'])
```

In [16]:
```
### BEGIN SOLUTION - DO NOT delete this comment for grading purposes
scikit_processing_job_status = processor.jobs[-1].describe()['ProcessingJ
### END SOLUTION - DO NOT delete this comment for grading purposes
print('Processing job status: {}'.format(scikit_processing_job_status))
```

```
Processing job status: InProgress
```

Review the created processing job in the AWS console.

Instructions:

- open the link
- notice that you are in the section `Amazon SageMaker` -> `Processing jobs`
- check the name of the processing job, its status and other available information

In [17]:
```
from IPython.core.display import display, HTML

display(HTML('<b>Review <a target="blank" href="https://console.aws.amazo
```

Review processing job

Wait for about 5 minutes to review the CloudWatch Logs. You may open the file src/prepare_data.py again and examine the outputs of the code in the CloudWatch logs.

In [18]:
```
from IPython.core.display import display, HTML

display(HTML('<b>Review <a target="blank" href="https://console.aws.amazo
```

Review CloudWatch logs after about 5 minutes

After the completion of the processing job you can also review the output in the S3 bucket.

In [19]:
```
from IPython.core.display import display, HTML

display(HTML('<b>Review <a target="blank" href="https://s3.console.aws.am
```

Review S3 output data after the processing job has completed

Wait for the processing job to complete.

This cell will take approximately 15 minutes to run.

In [20]:
```
%%time

running_processor = sagemaker.processing.ProcessingJob.from_processing_na
    processing_job_name=scikit_processing_job_name,
    sagemaker_session=sess
)

running_processor.wait(logs=False)
```

```
.....................................................!CPU times: u
ser 250 ms, sys: 42.2 ms, total: 292 ms
Wall time: 5min 5s
```

Please wait until ^^ Processing Job ^^ completes above

Inspect the transformed and balanced data in the S3 bucket.

In [21]:
```
processing_job_description = running_processor.describe()

output_config = processing_job_description['ProcessingOutputConfig']
for output in output_config['Outputs']:
    if output['OutputName'] == 'sentiment-train':
        processed_train_data_s3_uri = output['S3Output']['S3Uri']
    if output['OutputName'] == 'sentiment-validation':
        processed_validation_data_s3_uri = output['S3Output']['S3Uri']
    if output['OutputName'] == 'sentiment-test':
        processed_test_data_s3_uri = output['S3Output']['S3Uri']

print(processed_train_data_s3_uri)
print(processed_validation_data_s3_uri)
print(processed_test_data_s3_uri)
```

```
s3://sagemaker-us-east-1-504458356961/sagemaker-scikit-learn-2022-08-29-20-16-18-304/output/sentiment-train
s3://sagemaker-us-east-1-504458356961/sagemaker-scikit-learn-2022-08-29-20-16-18-304/output/sentiment-validation
s3://sagemaker-us-east-1-504458356961/sagemaker-scikit-learn-2022-08-29-20-16-18-304/output/sentiment-test
```

In [22]:
```
!aws s3 ls $processed_train_data_s3_uri/
```

```
2022-08-29 20:30:16    4890076 part-algo-1-womens_clothing_ecommerce_reviews.tsv
```

In [23]:
```
!aws s3 ls $processed_validation_data_s3_uri/
```

```
2022-08-29 20:30:17     264236 part-algo-1-womens_clothing_ecommerce_reviews.tsv
```

In [24]:
```
!aws s3 ls $processed_test_data_s3_uri/
```

```
2022-08-29 20:30:17     273558 part-algo-1-womens_clothing_ecommerce_reviews.tsv
```

Copy the data into the folder `balanced`.

In [25]:
```
!aws s3 cp $processed_train_data_s3_uri/part-algo-1-womens_clothing_ecomm
!aws s3 cp $processed_validation_data_s3_uri/part-algo-1-womens_clothing_
!aws s3 cp $processed_test_data_s3_uri/part-algo-1-womens_clothing_ecomme
```

```
download: s3://sagemaker-us-east-1-504458356961/sagemaker-scikit-learn-2022-08-29-20-16-18-304/output/sentiment-train/part-algo-1-womens_clothing_ecommerce_reviews.tsv to balanced/sentiment-train/part-algo-1-womens_clothing_ecommerce_reviews.tsv
download: s3://sagemaker-us-east-1-504458356961/sagemaker-scikit-learn-2022-08-29-20-16-18-304/output/sentiment-validation/part-algo-1-womens_clothing_ecommerce_reviews.tsv to balanced/sentiment-validation/part-algo-1-womens_clothing_ecommerce_reviews.tsv
download: s3://sagemaker-us-east-1-504458356961/sagemaker-scikit-learn-2022-08-29-20-16-18-304/output/sentiment-test/part-algo-1-womens_clothing_ecommerce_reviews.tsv to balanced/sentiment-test/part-algo-1-womens_clothing_ecommerce_reviews.tsv
```

Review the training, validation and test data outputs:

In [26]:
```
!head -n 5 ./balanced/sentiment-train/part-algo-1-womens_clothing_ecommer
```

```
review_id       sentiment       label_id        input_ids
review_body     date
3343    0       1       [0, 133, 1521, 1415, 2422, 11962, 804, 1437, 53,
11, 621, 1437, 5, 1468, 2551, 6162, 4, 888, 1437, 23, 513, 11, 5, 2272, 6
5, 1437, 5, 1468, 16, 430, 11, 5, 29874, 8, 2705, 9042, 131, 114, 24, 56,
70, 57, 5, 29874, 1468, 1437, 24, 74, 33, 57, 203, 35735, 4, 2, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1
, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]
The design looked super cute online  but in person  the material seemed c
heap. actually  at least in the green one  the material is different in t
he striped and solid sections; if it had all been the striped material  i
t would have been much nicer.   2022-08-29T20:24:48Z
1154    1       2       [0, 100, 2740, 209, 11, 127, 6097, 1836, 1437, 97
3, 1437, 8, 51, 2564, 6128, 6683, 4682, 5, 17507, 4, 13, 162, 1437, 939,
303, 106, 7, 28, 350, 3229, 11, 14, 443, 4, 939, 439, 7, 5, 1400, 8, 1381
, 15, 5, 974, 8, 51, 58, 350, 380, 11, 5, 13977, 4, 939, 109, 5940, 106,
600, 4, 51, 32, 157, 156, 1437, 939, 2638, 5, 10397, 8, 5, 5933, 21, 1969
, 4, 145, 195, 108, 246, 113, 51, 376, 7, 5, 795, 7451, 8, 14, 21, 2051,
19, 162, 4, 492, 106, 10, 860, 1437, 5952, 51, 173, 357, 13, 47, 4, 2, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]
"I ordered these in my typical size  26  and they fit everywhere perfectl
y except the butt. for me  i found them to be too tight in that area. i w
ent to the store and tried on the 27 and they were too big in the waist.
i do recommend them though. they are well made  i loved the wash and the
length was perfect. being 5'3"" they came to the lower ankle and that was
fine with me. give them a try  hopefully they work better for you."
2022-08-29T20:24:48Z
19353   -1      0       [0, 100, 21, 182, 6649, 7, 1325, 42, 3588, 19, 10
, 36069, 43805, 1437, 1326, 101, 22196, 219, 1104, 18309, 479, 192, 7391,
3493, 4, 5, 3588, 10199, 16, 182, 2579, 8, 939, 222, 860, 24, 15, 8, 24,
16, 10, 182, 2579, 2564, 1437, 9321, 19, 10, 410, 4140, 53, 127, 821, 521
2, 99, 10, 6162, 2129, 1318, 43805, 14, 1326, 101, 24, 21, 793, 8, 4924,
7, 16227, 142, 5, 4204, 1667, 33, 10, 1104, 22196, 219, 6572, 328, 6215,
47, 269, 240, 7, 1649, 110, 12952, 137, 3981, 7, 110, 8259, 916, 4, 12056
, 5, 1318, 34, 57, 8620, 10, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1]   I was very shocked to receive this dress
with a rusty zipper  looks like crusty white rust . see attached pictures
. the dress fabric is very nice and i did try it on and it is a very nice
fit  lined with a little stretch but my gosh what a cheap poor quality zi
pper that looks like it was old and exposed to moisture because the metal
parts have a white crusty substance! retailer you really need to check yo
ur merchandise before sending to your loyal customers. lately the quality
has been declining a    2022-08-29T20:24:48Z
1465    0       1       [0, 713, 3588, 16, 182, 11962, 8, 10698, 1341, 15
7, 4, 47, 218, 75, 33, 7, 3568, 10, 11689, 19, 24, 1437, 61, 16, 6344, 14
37, 8, 24, 34, 24897, 12189, 4, 5, 13422, 16, 14, 5, 10199, 15, 5, 299, 4
57, 16, 10, 410, 7735, 4, 24, 16, 182, 2016, 8, 13116, 4, 67, 1437, 5, 10
25, 34, 103, 6664, 11918, 8, 5, 1318, 473, 45, 2045, 7, 28, 5, 275, 4, 93
9, 437, 2396, 24, 600, 4, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1
, 1, 1, 1, 1, 1, 1, 1]       This dress is very cute and fits quite we
ll. you don't have to wear a bra with it  which is awesome  and it has de
lightful pockets. the downside is that the fabric on the top half is a li
ttle weird. it is very heavy and stiff. also  the inside has some fraying
and the quality does not seem to be the best. i'm keeping it though.
2022-08-29T20:24:48Z
```

In [27]:
```
!head -n 5 ./balanced/sentiment-validation/part-algo-1-womens_clothing_ec
```

```
review_id       sentiment       label_id        input_ids
review_body     date
15360   -1      0       [0, 1708, 1437, 24, 95, 630, 75, 2450, 62, 4, 78,
1437, 5, 9215, 73, 16526, 606, 23983, 12552, 114, 2468, 2829, 8, 24, 16,
182, 1202, 7, 888, 120, 24, 15, 111, 1169, 8296, 88, 24, 50, 6539, 24, 81
, 110, 471, 4, 8, 1437, 683, 15, 1437, 5, 299, 21, 10, 380, 41783, 8, 89,
21, 350, 203, 1468, 23, 5, 2576, 61, 156, 24, 356, 269, 35156, 4, 9574, 1
437, 939, 524, 2609, 42, 7, 28, 5, 403, 19, 98, 171, 9, 5, 1964, 939, 907
, 31, 6215, 12056, 4, 5, 1468, 16, 269, 11962, 1437, 53, 5, 3588, 965, 75
, 966, 5, 425, 1589, 1351, 4, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1
, 1, 1, 1]      But  it just doesn't measure up. first  the slip/lining c
omes unattached if pulled slightly and it is very difficult to actually g
et it on - either stepping into it or pulling it over your head. and  onc
e on  the top was a big snug and there was too much material at the botto
m which made it look really bulky. unfortunately  i am finding this to be
the case with so many of the items i buy from retailer lately. the materi
al is really cute  but the dress isn't worth the price / effort.
2022-08-29T20:24:48Z
7838    1       2       [0, 713, 5780, 16, 98, 3035, 328, 5, 6399, 16, 95
, 10, 410, 7082, 98, 645, 110, 4505, 1836, 8, 24, 40, 2564, 6683, 4, 2, 1
, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1
, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1
, 1, 1, 1]      This print is so cool! the shirt is just a little loose s
o order your usual size and it will fit perfectly.      2022-08-29T20:24:
48Z
19592   -1      0       [0, 100, 437, 4716, 1459, 1437, 8, 939, 1467, 215
9, 42, 14, 24, 74, 28, 10, 380, 23204, 111, 14, 16, 5, 356, 14, 16, 15266
, 15, 5, 1421, 4, 5, 23204, 16, 12272, 219, 1437, 5, 5397, 21, 98, 380, 8
, 35156, 939, 56, 3605, 442, 24, 4477, 235, 4, 24, 1326, 372, 15, 5, 1421
, 259, 1437, 53, 45, 15, 162, 4, 164, 124, 4, 98, 29673, 4567, 4, 5, 3195
, 12339, 16, 1528, 7, 99, 16, 2343, 8, 24, 34, 410, 22, 31670, 113, 50, 1
8459, 420, 5, 10762, 4, 98, 89, 16, 10, 5262, 828, 9, 4617, 4, 1374, 1437
, 45, 10, 372, 23204, 4, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1]        "I'm petite  and i knew buying this that it would be a bi
g sweater - that is the look that is portrayed on the model. the sweater
is scratchy  the neck was so big and bulky i had trouble making it lay ri
ght. it looks great on the model here  but not on me. going back. so bumm
ed. the color gray is true to what is shown and it has little ""holes"" o
r openings across the shoulders. so there is a tiny bit of detail. overal
l  not a great sweater."        2022-08-29T20:24:48Z
9833    0       1       [0, 713, 21, 95, 45, 34203, 13, 162, 5579, 13724,
196, 350, 203, 101, 10, 21592, 3588, 4, 939, 64, 75, 269, 109, 10, 12138,
5357, 13977, 98, 2532, 14, 74, 146, 42, 3588, 55, 16106, 13, 951, 1493, 4
, 939, 21, 6661, 7, 33, 7, 671, 142, 24, 1411, 157, 19, 6255, 1536, 8, 10
317, 328, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1
, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1]      This was just not flattering for me--looked too much like a mater
nity dress. i can't really do a belted waist so perhaps that would make t
his dress more versatile for someone else. i was sorry to have to return
because it goes well with sandals and boots!    2022-08-29T20:24:48Z
```

In [28]:
```
!head -n 5 ./balanced/sentiment-test/part-algo-1-womens_clothing_ecommerc
```

```
review_id       sentiment       label_id        input_ids
review_body     date
13528   -1      0       [0, 133, 23204, 16, 182, 3137, 24382, 8, 1415, 20
5, 5, 78, 86, 939, 1381, 24, 15, 1437, 959, 5, 1468, 16, 7174, 8, 2829, 1
92, 149, 4, 67, 1437, 71, 65, 3568, 24, 2198, 13596, 66, 8, 156, 5, 251,
21764, 11789, 7, 3568, 396, 14784, 11, 227, 349, 3568, 4, 71, 5, 371, 86,
939, 5328, 24, 1437, 939, 2967, 10, 739, 4683, 11, 5, 4709, 6195, 405, 14
37, 144, 533, 528, 7, 5, 10079, 14784, 4, 939, 74, 45, 5940, 42, 23204, 4
, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]   The sweater is ve
ry comfy and looked good the first time i tried it on  however the materi
al is thin and slightly see through. also  after one wear it completely s
tretched out and made the long sleeves awkward to wear without washing in
between each wear. after the third time i wore it  i discovered a large h
ole in the armpit  most likely due to the excessive washing. i would not
recommend this sweater. 2022-08-29T20:24:48Z
1089    1       2       [0, 100, 1467, 939, 770, 42, 299, 5, 1151, 939, 7
94, 24, 1437, 98, 2500, 127, 2813, 8458, 24, 439, 454, 24, 439, 15, 1392,
328, 24, 10698, 16467, 11, 127, 2340, 1836, 8, 939, 657, 14, 24, 416, 34,
10, 22220, 6013, 299, 4, 939, 120, 33391, 8378, 939, 3568, 24, 12, 206, 1
368, 417, 11, 2242, 354, 429, 28, 127, 92, 213, 12, 560, 328, 2, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1
, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1] I knew i wanted this top the moment i saw it  so
onto my wishlist it went until it went on sale! it fits beautifully in my
normal size and i love that it already has a coordinating tank top. i get
compliments whenever i wear it- think hd in paris might be my new go-to!
2022-08-29T20:24:48Z
15824   1       2       [0, 16587, 5, 3793, 10199, 8, 1365, 2564, 9, 42,
3588, 4, 939, 437, 4343, 14, 24, 473, 492, 10, 11602, 9, 65, 18, 1955, 8,
630, 75, 95, 6713, 101, 10, 14072, 4, 939, 303, 5, 2440, 7, 28, 144, 3420
3, 8, 21, 816, 19, 23256, 23, 184, 8, 303, 14, 5, 6012, 6215, 22, 34108,
1409, 113, 9219, 7494, 851, 24, 10, 1086, 92, 356, 4, 2, 1, 1, 1, 1, 1, 1
, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1
, 1, 1, 1, 1]   "Love the soft fabric and easy fit of this dress. i'm ple
ased that it does give a hint of one's figure and doesn't just hang like
a sack. i found the blue to be most flattering and was playing with styli
ng at home and found that the wider retailer ""tabby"" leather belt gave
it a whole new look."   2022-08-29T20:24:48Z
19232   -1      0       [0, 100, 2638, 209, 9304, 8, 3584, 106, 11, 130,
8089, 4, 101, 277, 37102, 1437, 624, 10, 367, 688, 1437, 519, 10610, 106,
117, 55, 87, 10, 891, 9, 498, 1437, 5, 10199, 11, 5, 42613, 443, 554, 7,
3568, 7174, 8, 80, 15029, 2226, 6538, 4, 939, 21, 45, 543, 15, 5, 9304, 1
11, 95, 5328, 106, 7, 5, 558, 111, 8, 516, 16380, 106, 4, 98, 6770, 1437,
25, 939, 269, 222, 657, 5, 9304, 4, 746, 3844, 9, 418, 4, 2, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1
, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]     I loved these pants and p
urchased them in three colors. like another reviewer  within a few weeks
having worn them no more than a couple of times  the fabric in the crotch
area started to wear thin and two pairs developed holes. i was not hard o
n the pants - just wore them to the office - and line dried them. so disa
ppointing  as i really did love the pants. total waste of money.
2022-08-29T20:24:48Z
```

3. Query the Feature Store

In addition to transforming the data and saving in S3 bucket, the processing job populates the feature store with the transformed and balanced data. Let's query this data using Amazon Athena.

3.1. Export training, validation, and test datasets from the Feature Store

Here you will do the export only for the training dataset, as an example.

Use `athena_query()` function to create an Athena query for the defined above Feature Group. Then you can pull the table name of the Amazon Glue Data Catalog table which is auto-generated by Feature Store.

In [29]:
```
feature_store_query = feature_group.athena_query()

feature_store_table = feature_store_query.table_name

query_string = """
    SELECT date,
        review_id,
        sentiment,
        label_id,
        input_ids,
        review_body
    FROM "{}"
    WHERE split_type='train'
    LIMIT 5
""".format(feature_store_table)

print('Glue Catalog table name: {}'.format(feature_store_table))
print('Running query: {}'.format(query_string))
```

```
Glue Catalog table name: reviews-feature-group-1661803330-1661804658
Running query:
    SELECT date,
        review_id,
        sentiment,
        label_id,
        input_ids,
        review_body
    FROM "reviews-feature-group-1661803330-1661804658"
    WHERE split_type='train'
    LIMIT 5
```

Configure the S3 location for the query results. This allows us to re-use the query results for future queries if the data has not changed. We can even share this S3 location between team members to improve query performance for common queries on data that does not change often.

In [30]:
```
output_s3_uri = 's3://{}/query_results/{}/'.format(bucket, feature_store_
print(output_s3_uri)
```

```
s3://sagemaker-us-east-1-504458356961/query_results/reviews-feature-store
-1661803330/
```

Exercise 4

Query the feature store.

Instructions: Use `feature_store_query.run` function passing the constructed above query string and the location of the output S3 bucket.

```
feature_store_query.run(
    query_string=..., # query string
    output_location=... # location of the output S3 bucket
)
```

In [31]:
```
feature_store_query.run(
    ### BEGIN SOLUTION - DO NOT delete this comment for grading purposes
    query_string=query_string, # Replace None
    output_location=output_s3_uri # Replace None
    ### END SOLUTION - DO NOT delete this comment for grading purposes
)

feature_store_query.wait()
```

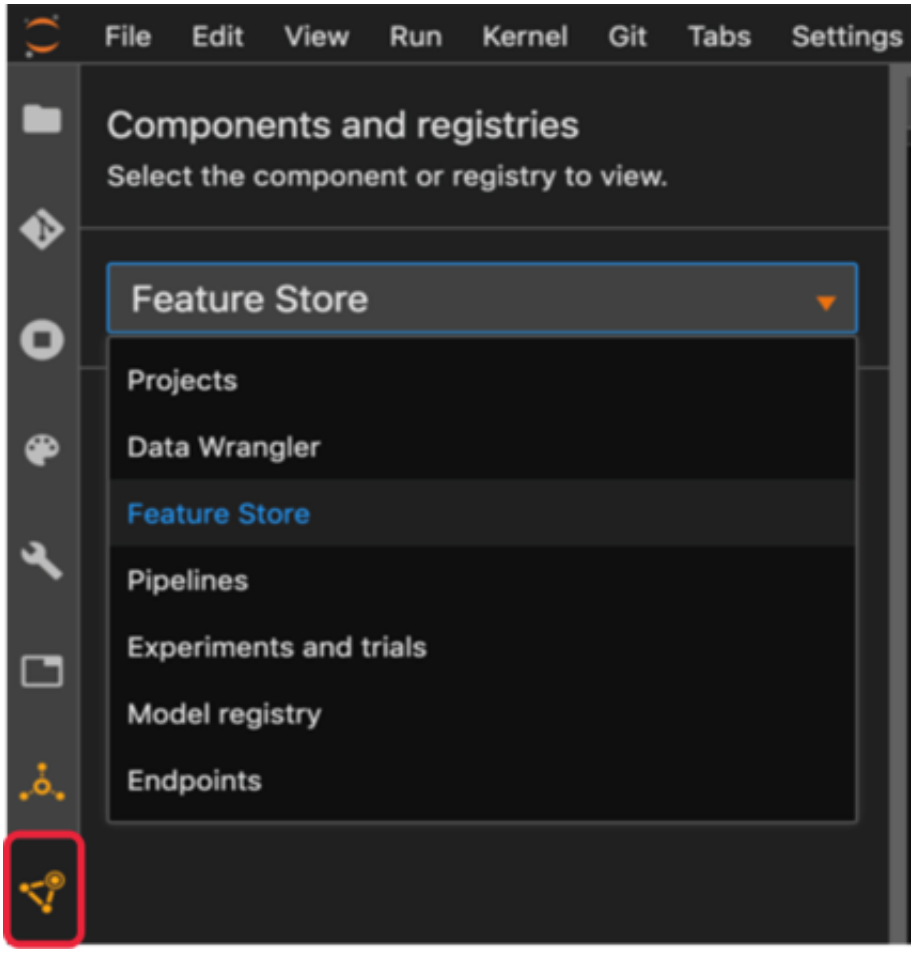
In [32]:
```
import pandas as pd
pd.set_option("max_colwidth", 100)

df_feature_store = feature_store_query.as_dataframe()
df_feature_store
```

Out[32]:

	date	review_id	sentiment	label_id	input_ids	review_body
0	2022-08-29T20:24:48Z	7558	1	2	[0, 41541, 1318, 1437, 1254, 1437, 8, 939, 101, 5, 4600, 19780, 847, 36, 11655, 157, 15, 42, 308...	Nice quality details and i like the looser cut (works well on this blouse).
1	2022-08-29T20:24:48Z	3803	0	1	[0, 35703, 10, 1256, 475, 4791, 548, 6399, 14, 888, 2564, 127, 36, 3340, 45314, 43, 3065, 1368, ...	Finally a pretty maeve shirt that actually fit my (apparently) giant hulk arms.. but it's so flo...
2	2022-08-29T20:24:48Z	3316	1	2	[0, 597, 2629, 6683, 328, 2422, 3793, 8, 157, 12, 7078, 328, 657, 5, 155, 73, 306, 21764, 8, 124...	Fits perfectly! super soft and well-made! love the 3/4 sleeves and back drop. i can expect i wil...
3	2022-08-29T20:24:48Z	4699	1	2	[0, 100, 657, 42, 23204, 4, 24, 18, 3279, 1437, 3473, 1437, 8, 3793, 4, 2579, 1318, 350, 4, 2, 1...	I love this sweater. it's warm comfortable and soft. nice quality too.
4	2022-08-29T20:24:48Z	16894	0	1	[0, 100, 101, 42, 299, 53, 9574, 939, 40, 28, 3357, 4, 24, 18, 169, 350, 251, 4, 5, 3195, 16, 12...	I like this top but unfortunately i will be returning. it's way too long. the color is pretty as...

Review the Feature Store in SageMaker Studio



3.2. Export TSV from Feature Store

Save the output as a TSV file:

In [33]:
```
df_feature_store.to_csv('./feature_store_export.tsv',
                        sep='\t',
                        index=False,
                        header=True)
```

In [34]:
```
!head -n 5 ./feature_store_export.tsv
```

```
date    review_id       sentiment       label_id        input_ids
review_body
2022-08-29T20:24:48Z    7558    1       2       [0, 41541, 1318, 1437, 12
54, 1437, 8, 939, 101, 5, 4600, 19780, 847, 36, 11655, 157, 15, 42, 3089,
11556, 322, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1
, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1
, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]  Nice quality  details  and i like
the looser cut (works well on this blouse).
2022-08-29T20:24:48Z    3803    0       1       [0, 35703, 10, 1256, 475,
4791, 548, 6399, 14, 888, 2564, 127, 36, 3340, 45314, 43, 3065, 1368, 162
60, 3701, 7586, 53, 24, 18, 98, 3041, 219, 73, 32123, 219, 939, 56, 7, 67
1, 24, 4, 42, 429, 95, 28, 10, 936, 19, 145, 10, 2514, 6429, 1437, 53, 5,
7400, 10199, 156, 162, 619, 101, 939, 21, 2498, 10, 10178, 8, 156, 162, 3
56, 158, 1178, 2671, 87, 939, 21, 198, 127, 13977, 73, 7903, 1437, 190, 7
7, 9320, 66, 19, 2084, 6149, 1033, 4, 939, 802, 59, 667, 7, 1836, 159, 53
, 938, 75, 686, 114, 14, 74, 190, 6136, 5, 7400, 10199, 936, 8, 202, 989,
929, 11, 5, 3701, 4, 9, 768, 939, 64, 75, 860, 2, 1, 1, 1, 1, 1, 1, 1]
Finally a pretty maeve shirt that actually fit my (apparently) giant hulk
arms.. but it's so flowy/swingy i had to return it. this might just be a
problem with being a larger lady  but the excess fabric made me feel like
i was wearing a tent and made me look 10x bigger than i was around my wai
st/hips  even when balanced out with leggings. i thought about trying to
size down but wasn't sure if that would even solve the excess fabric prob
lem and still leave room in the arms. of course i can't try
2022-08-29T20:24:48Z    3316    1       2       [0, 597, 2629, 6683, 328,
2422, 3793, 8, 157, 12, 7078, 328, 657, 5, 155, 73, 306, 21764, 8, 124, 1
874, 4, 939, 64, 1057, 939, 40, 3568, 10, 319, 11, 42, 567, 1136, 73, 314
21, 191, 11, 6056, 4, 2200, 5940, 328, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1
, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1
, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]
Fits perfectly! super soft and well-made! love the 3/4 sleeves and back d
rop. i can expect i will wear a lot in this coming fall/winter season in
ca. highly recommend!
2022-08-29T20:24:48Z    4699    1       2       [0, 100, 657, 42, 23204,
4, 24, 18, 3279, 1437, 3473, 1437, 8, 3793, 4, 2579, 1318, 350, 4, 2, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1
, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1
, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1]      I love this sweater. it's warm  comfortab
le  and soft. nice quality too.
```

Upload TSV to the S3 bucket:

In [35]:
```
!aws s3 cp ./feature_store_export.tsv s3://$bucket/feature_store/feature_
```

```
upload: ./feature_store_export.tsv to s3://sagemaker-us-east-1-5044583569
61/feature_store/feature_store_export.tsv
```

Check the file in the S3 bucket:

In [36]:
```
!aws s3 ls --recursive s3://$bucket/feature_store/feature_store_export.ts
```

```
2022-08-29 20:33:26       3665 feature_store/feature_store_export.tsv
```

3.3. Check that the dataset in the Feature Store is balanced by sentiment

Now you can setup an Athena query to check that the stored dataset is balanced by the target class `sentiment`.

Exercise 5

Write an SQL query to count the total number of the reviews per `sentiment` stored in the Feature Group.

Instructions: Pass the SQL statement of the form

```
SELECT category_column, COUNT(*) AS new_column_name
FROM table_name
GROUP BY category_column
```

into the variable `query_string_count_by_sentiment`. Here you would need to use the column `sentiment` and give a name `count_reviews` to the new column with the counts.

In [37]:
```
feature_store_query_2 = feature_group.athena_query()

# Replace all None
### BEGIN SOLUTION - DO NOT delete this comment for grading purposes
query_string_count_by_sentiment = """
SELECT sentiment, COUNT(*) AS count_reviews
FROM "{}"
GROUP BY sentiment
""".format(feature_store_table)
### END SOLUTION - DO NOT delete this comment for grading purposes
```

Exercise 6

Query the feature store.

Instructions: Use `run` function of the Feature Store query, passing the new query string `query_string_count_by_sentiment`. The output S3 bucket will remain unchanged. You can follow the example above.

In [38]:
```
feature_store_query_2.run(
    ### BEGIN SOLUTION - DO NOT delete this comment for grading purposes
    query_string=query_string_count_by_sentiment, # Replace None
    output_location=output_s3_uri # Replace None
    ### END SOLUTION - DO NOT delete this comment for grading purposes
)

feature_store_query_2.wait()

df_count_by_sentiment = feature_store_query_2.as_dataframe()
df_count_by_sentiment
```

Out[38]:

	sentiment	count_reviews
0	0	2051
1	1	2051
2	-1	2051

Exercise 7

Visualize the result of the query in the bar plot, showing the count of the reviews by sentiment value.

Instructions: Pass the resulting data frame `df_count_by_sentiment` into the `barplot` function of the `seaborn` library.

```
sns.barplot(
    data=...,
    x='...',
    y='...',
    color="blue"
)
```

In [39]:
```
import seaborn as sns

sns.barplot(
    ### BEGIN SOLUTION - DO NOT delete this comment for grading purposes
    data=df_count_by_sentiment, # Replace None
    x='sentiment', # Replace None
    y='count_reviews', # Replace None
    ### END SOLUTION - DO NOT delete this comment for grading purposes
    color="blue"
)
```

Out[39]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7fd1afc93dd0>
```

Upload the notebook and `prepare_data.py` file into S3 bucket for grading purposes.

Note: you may need to save the file before the upload.

In [40]:
```
!aws s3 cp ./C2_W1_Assignment.ipynb s3://$bucket/C2_W1_Assignment_Learner
!aws s3 cp ./src/prepare_data.py s3://$bucket/src/C2_W1_prepare_data_Lear
```

```
upload: ./C2_W1_Assignment.ipynb to s3://sagemaker-us-east-1-504458356961
/C2_W1_Assignment_Learner.ipynb
upload: src/prepare_data.py to s3://sagemaker-us-east-1-504458356961/src/
C2_W1_prepare_data_Learner.py
```

Please go to the main lab window and click on `Submit` button (see the `Finish the lab` section of the instructions).

In [ ]: