

## aml4-q2

December 12, 2023

We are considering a small dataset of 320 line which classifies a sentence as truthful or deceptive. The pre trained model we have used here is BERT. BERT is based on the transformer architecture, which was introduced in the paper “Attention is All You Need” by Vaswani et al. The transformer architecture uses self-attention mechanisms to capture contextual information from both left and right contexts. it has 12 transformer layer and 12 attention heads.

```
[1]: import pandas as pd
```

```
[12]: from sklearn.model_selection import train_test_split
from collections import Counter
path = '/content/review_dataset_train.txt'
# Read dataset from file
with open(path, 'r') as file:
    lines = file.readlines()

# Extract labels and sentences
data = [line.strip().split(' ', 1) for line in lines]
labels, sentences = zip(*data)

# Count the total sentences and each label
total_sentences = len(sentences)
label_counts = Counter(labels)
print('Total no of lines:', total_sentences)
print('Label count for each:', label_counts)
```

Total no of lines: 320

Label count for each: Counter({'deceptive': 164, 'truthful': 156})

```
[13]: # Display total counts
print("\nTotal Counts:")
print(f"Total Sentences: {total_sentences}")
for label, count in label_counts.items():
    print(f"{label} Count: {count}")
```

Total Counts:

Total Sentences: 320

truthful Count: 156

deceptive Count: 164

```
[14]: from sklearn.model_selection import train_test_split

# Read dataset from file
with open(path, 'r') as file:
    lines = file.readlines()

# Extract labels and sentences
data = [line.strip().split(' ', 1) for line in lines]
labels, sentences = zip(*data)

# Split the dataset into training and test sets (80:20 ratio)
train_sentences, test_sentences, train_labels, test_labels = train_test_split(
    sentences, labels, test_size=0.2, random_state=42
)
```

```
[23]: print('Length of Train',len(train_sentences))
      print('Length of Test',len(test_sentences))
```

Length of Train 256

Length of Test 64

```
[15]: # Display the first 5 lines of the training set
      print("Training Set:")
      for label, sentence in zip(train_labels[:5], train_sentences[:5]):
          print(f"{label}: {sentence}")
```

Training Set:

truthful: My family booked five nights at the Omni Chicago for our upcoming spring break. We took advantage of a pre-paid, non-cancellable offer. We subsequently received a 'Bed & Breakfast' offer via email and immediately called the hotel to inquire whether we could switch our reservation (same dates, different offer). We called the 800' number and the customer services representative was so obnoxiously condescending and ignorant, we simply hung up. We had been looking forward to our vacation, but now I am not sure I made the right choice with this hotel. I should have paid closer attention to the reviews - the Ritz, the Four Seasons, and the Peninsula would have never treated us this way. I will post again after our stay on April 17th.

deceptive: While travelling for business I had my family join me and we stayed at the Fairmont Chicago, Millennium Park. It is breathtakingly beautiful and plush. The rates for rooms, though not the cheapest choice are quite reasonable for such fabulous amenities. The room decor is elegant modern and my wife says the spa is divine! Guests of this hotel are able to live in luxury - every room is equipped with complementary coffee, access to high speed internet, bathrobes, and 42 inch flat screen televisions. Enjoy the gorgeous city views and live the good life at the Fairmont Chicago, Millennium Park.

truthful: My sister and I went to Chicago for a weekend and decided to stay at this hotel because I have a friend who works at Homewood Suites here in Cleveland so we got a discount. The valets were really friendly, you get in and out service when you pay for parking with the hotel. Check-in is on the 6th floor because there are restaurants beneath the hotel at street level. That also made finding the hotel tricky because I expected a groove to pull into and it turned out that I just had to pull over to the right side of the street. The lady at the front desk was friendly and informative. Our room was very clean. The fact that it is an extended stay hotel was nice when it came to bringing home leftovers from restaurants, storing and reheating food. The complimentary breakfast was great! They have real food such as eggs, bacon, french toast, cereal, bagels, juice, milk, and yogurt. The hotel is in walking distance from everything and there is a trolley that goes to Navy Pier and passes by Michigan Avenue. They also have free WiFi. Overall, I would definitely stay here again and I would recommend it to others.

deceptive: I stayed in the Sofitel Chicago Water Tower hotel with my husband and two kids last weekend, and will definitely use this hotel again for future travels. The spacious rooms with modern, innovative furniture and comfortable beds was a great contrast to the small, dismal rooms and rock-hard beds that I have found at other hotel chains. The bathrooms were neat, tidy, and well-stocked with towels and personal needs such as toilet paper, shampoo, and soap. The carpet was clean (not stained and dingy like some hotels), the room decor was astounding, and the view from our window was wonderful. I enjoyed every aspect of this hotel stay, from the courteous staff at check-in to the ease of requesting room service and checking out. Thanks for making me feel right at home!

deceptive: What a disappointment this hotel was. First, the checkin desk was understaffed and we had to wait forever to get through the line to check in. When we finally got up to our room, we opened the door and found someone else was already in it! Yes, the front desk clerk actually checked us into an occupied room and basically gave us a key to someone else's room. We went back to the clerk who checked us in (skipping the still present line, which got us annoyed stares from the other people waiting) and told him the problem. However, he was already helping another guest checkin and made us wait until he was done with that person before helping us. He did apologize for the mix up and then went into the back room to find out what to do. This took a \*very\* long time. It was a good 10-15 minutes until he came back. He said there were no more rooms available of the type we had booked and the only ones they had left were on a lower floor with no view. The hotel offered us a discount off the regular price for these, but when we got to this new room, we found out why these rooms hadn't been taken. The room was overlooking the loading dock at the back of the hotel. We briefly considered just leaving and going to another hotel, but it was late already so we just decided to make do. That was, of course, until the garbage truck came around 6am and made all sorts of noise picking up the dumpsters and dropping them back down. Then somewhere around 7:30am the delivery trucks started showing up making that beep-beep backup sound and all sorts of other noise. So much for a good night's sleep. I suppose I should say the room itself looked nice, but with all the noise from the loading dock, who cares. We gave up

and checked out that morning to find another hotel. Definitely not going back here ever again.

```
[16]: # Display the first 5 lines of the test set
print("\nTest Set:")
for label, sentence in zip(test_labels[:5], test_sentences[:5]):
    print(f"{label}: {sentence}")
```

Test Set:

deceptive: I stayed at the Hard Rock hotel in Chicago last year and can say i was not satisfied with my experience. I feel that it didn't live up to what was advertised and the accommodations were sub-par at best. I had the chance to catch a show and the music was too loud and the sound guy's work resembled that of an amateur. My personal rating is 2/5 stars and i would not recommend this hotel to a friend.

truthful: My husband and I stayed overnight at the Affinia. Prior reviews led us to believe this was a quality hotel; not true. Since it was a very special occasion, and also because we were only staying overnight, we didn't let the shabbiness and poor service of the hotel ruin our weekend. However, this would have qualified as one of the poorest hotel experiences. I've ever had. First, the smells: The linens and pillows on our bed smelled moldy and mildewy. The odors were not mild, and I found it pretty repulsive. I couldn't wait to jump out of bed in the morning, to get away from the odors. Hard to sleep and certainly didn't want to be in the bed any longer than absolutely necessary. I didn't discover how disgusting the bed smelled until around midnight when we got into bed. At that point, I chose not to contact housekeeping because (a) I was exhausted, and (b) I didn't want to upset my husband, who had fallen immediately to sleep after a long day of sightseeing. Also, the bathroom smelled like vinegar. Perhaps this is the cleaning agent used by Affinia, but the odor should have dissipated over the course of our stay and it never did. Incidentally, the bathroom floor tiles and wallpaper looked like they had seen better days; not a well-maintained property. Second, the elevators: If you stay above the first few floors, there are only two elevators that run from ground level to the rooftop c-view bar. With so many floors to service, and so many people going up to the c-view bar on a saturday, these elevators took forever to come, and were often full they did. Annoying. Also, on one occasion, there were several bottles (water and soda) left in the elevator; clearly, the maintenance crew doesn't inspect and clean the common areas frequently. Finally, and perhaps the most disappointing was the service when we arrived. First, when we booked, we asked for as quiet a room as possible. The hotel is located near Northwestern Memorial Hospital. But, its proximity to Michigan Ave., and the lack of accomodations elsewhere (due to the Transformers movie filming and a big international convention taking place the same week-end), led us to stay here anyway. At check-in, the clerk noted our request, and said our room was quiet room since it was not 'next to the elevators.' When we arrived at our room, we discovered that we were across from the elevators. Obviously, she was technically correct that we weren't next to the elevators, but it was still so close (15-20 feet) that we

could hear the elevator noise. Also, since it was a special occasion, we asked if we could upgrade. The disingenuous, snarky, condescending clerk told us that checking in early was an upgrade. Clearly, the woman has not travelled enough to know that in many quality hotels, this is a matter of routine, not exception. Pretty awful experience all the way around. Other reviewers must have very low expectations. Can't explain any of the good reviews otherwise.

deceptive: I accompanied my husband on a business trip to Chicago back in September and we stayed at the Millennium Knickerbocker Hotel. The place was absolutely oozing with old-fashioned charm! Also, there were plenty of shopping opportunities within walking distance, so I shopped while my husband worked. It just doesn't get any better than that! I loved the service we received and the ambiance of that grand old hotel, and I'd definitely stay there again!

deceptive: We recently completed our first visit to Chicago and were pleasantly surprised by the Affinia Chicago hotel where we stayed. The hotel has gone to great lengths to offer extra services to their guests, including individual pillow choices, complimentary kits with items designed to enhance the city visit, and fine dining in two on-site restaurants, one of them roof-top with amazing city views. I would recommend this hotel to anyone visiting Chicago.

deceptive: It was my first time to visit Chicago and I would say, i'd definitely go back, not just because of the Air and Water Show that we really enjoyed, but most importantly, because of the hotel that we stayed in. The Millennium Knickerbocker is not just accommodating to their human guests, but with pets too. At first I was reluctant that they will give us accommodations because of my pet dog, but I was surprised it turned out the other way. I've experienced being denied reservations because of my dog and that's really frustrating, it's just nice that there still are other hotels that are pet friendly. I would definitely recommend this hotel to my friends if they plan to visit Chicago.

```
[17]: import torch
      from transformers import BertTokenizer, BertForSequenceClassification, AdamW
      from torch.utils.data import DataLoader, TensorDataset
      from sklearn.model_selection import train_test_split
      from sklearn.metrics import accuracy_score
      from tqdm import tqdm
```

```
[18]: from sklearn.preprocessing import LabelEncoder
      # Assume train_labels and test_labels are lists of strings
      label_encoder = LabelEncoder()
      train_labels = label_encoder.fit_transform(train_labels)
      test_labels = label_encoder.transform(test_labels)

      # Convert labels to PyTorch tensors
      train_labels = torch.tensor(train_labels)
      test_labels = torch.tensor(test_labels)
```

```
[19]: # Load BERT tokenizer and model
      tokenizer = BertTokenizer.from_pretrained('bert-base-uncased')
```

```

model = BertForSequenceClassification.from_pretrained('bert-base-uncased',
    ↪num_labels=2)

# Tokenize the input data
train_encodings = tokenizer(train_sentences, truncation=True, padding=True,
    ↪max_length=128, return_tensors='pt')
test_encodings = tokenizer(test_sentences, truncation=True, padding=True,
    ↪max_length=128, return_tensors='pt')

# Convert labels to PyTorch tensors
train_labels = torch.tensor(train_labels)
test_labels = torch.tensor(test_labels)

# Create DataLoader for training and test sets
train_dataset = TensorDataset(train_encodings['input_ids'],
    ↪train_encodings['attention_mask'], train_labels)
test_dataset = TensorDataset(test_encodings['input_ids'],
    ↪test_encodings['attention_mask'], test_labels)

train_loader = DataLoader(train_dataset, batch_size=8, shuffle=True)
test_loader = DataLoader(test_dataset, batch_size=8, shuffle=False)

# Set up GPU/CPU and optimizer
device = torch.device('cuda' if torch.cuda.is_available() else 'cpu')
model.to(device)
optimizer = AdamW(model.parameters(), lr=2e-5)

# Training loop
num_epochs = 3
for epoch in range(num_epochs):
    model.train()
    for batch in tqdm(train_loader, desc=f'Epoch {epoch + 1}/{num_epochs}'):
        inputs = {'input_ids': batch[0].to(device),
            'attention_mask': batch[1].to(device),
            'labels': batch[2].to(device)}
        optimizer.zero_grad()
        outputs = model(**inputs)
        loss = outputs.loss
        loss.backward()
        optimizer.step()

```

Some weights of BertForSequenceClassification were not initialized from the model checkpoint at bert-base-uncased and are newly initialized:

```
['classifier.weight', 'classifier.bias']
```

You should probably TRAIN this model on a down-stream task to be able to use it for predictions and inference.

<ipython-input-19-a25db235a46c>:10: UserWarning: To copy construct from a

tensor, it is recommended to use `sourceTensor.clone().detach()` or `sourceTensor.clone().detach().requires_grad_(True)`, rather than `torch.tensor(sourceTensor)`.

```
train_labels = torch.tensor(train_labels)
<ipython-input-19-a25db235a46c>:11: UserWarning: To copy construct from a
tensor, it is recommended to use sourceTensor.clone().detach() or
sourceTensor.clone().detach().requires_grad_(True), rather than
torch.tensor(sourceTensor).
test_labels = torch.tensor(test_labels)
/usr/local/lib/python3.10/dist-packages/transformers/optimization.py:411:
FutureWarning: This implementation of AdamW is deprecated and will be removed in
a future version. Use the PyTorch implementation torch.optim.AdamW instead, or
set `no_deprecation_warning=True` to disable this warning
warnings.warn(
Epoch 1/3: 100%|      | 32/32 [06:32<00:00, 12.26s/it]
Epoch 2/3: 100%|      | 32/32 [06:06<00:00, 11.47s/it]
Epoch 3/3: 100%|      | 32/32 [06:31<00:00, 12.23s/it]
```



```
[20]: # Evaluation
model.eval()
test_predictions = []
with torch.no_grad():
    for batch in tqdm(test_loader, desc='Testing'):
        inputs = {'input_ids': batch[0].to(device),
                  'attention_mask': batch[1].to(device),
                  'labels': None}
        outputs = model(**inputs)
        logits = outputs.logits
        predictions = torch.argmax(logits, dim=1).cpu().numpy()
        test_predictions.extend(predictions)
```

```
Testing: 100%|      | 8/8 [00:32<00:00, 4.01s/it]
```

```
[21]: # Calculate accuracy
accuracy = accuracy_score(test_labels.numpy(), test_predictions)
print(f'Test Accuracy: {accuracy}')
```

Test Accuracy: 0.84375

The Test Accuracy is ~ 85% which is fairly good. Other improvements that can be made to make the

1. Increase the dataset size, here we have considered a small dataset of 320 lines
2. Hyperparameter tuning: Experiment with weights, epochs during training the data
3. Add regularization and dropouts
4. We can also experiment with different pre trained models such as gpt, XLNet
5. Make some changes and experiment with model by freezing few layers and add additional layers
6. Maintain balanced dataset, i.e. make sure it isn't skewed