Sentiment Analysis of Bank Customer Reviews using Bidirectional Encoder Representations from Transformer

Thanapoom Phatthanaphan
Deep Learning, Computer Science Department
Stevens Institute of Technology

Project Member

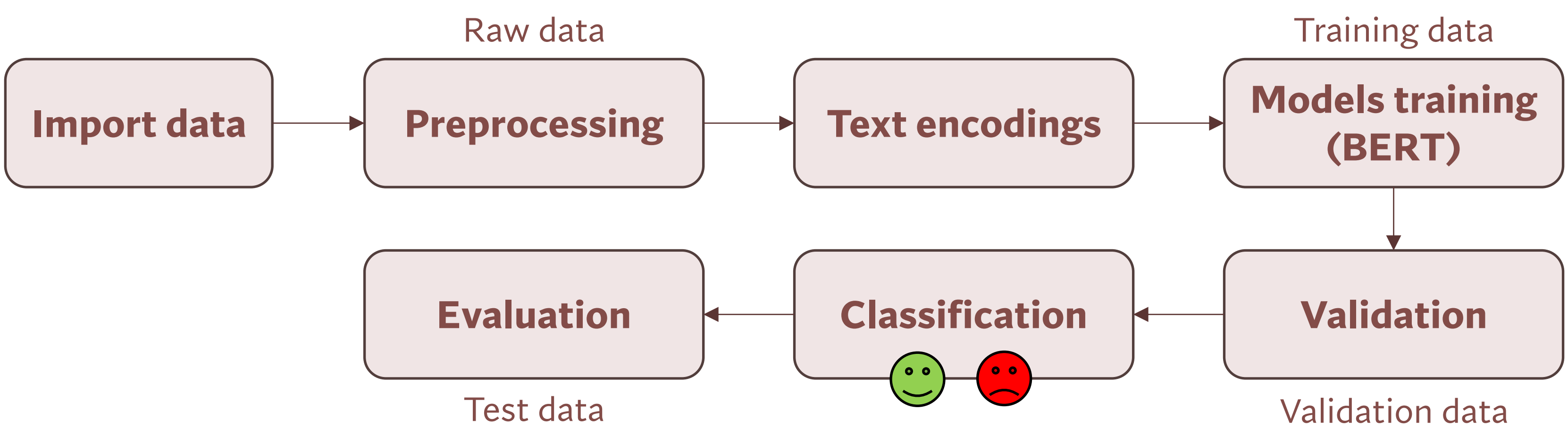
Thanapoom Phatthanphan

CWID 20011296
Degree Master’s Degree in Computer Science

Overview of the project

Build a **Sentiment analysis system** using **Bidirectional Encoder Representations from Transformer (BERT)** to analyze bank customer reviews to **accurately and efficiently classify** more than 10,000 recent **customer reviews** from 48 US banks as either **positive or negative**.

Steps of implementation



Dataset

More than 10,000 recent customer reviews from 48 US Banks

Public dataset
https://www.kaggle.com/datasets/trainingdatapro/20000-customers-reviews-on-banks/data

	author	date	location	bank	star	text	like
0	Kyle	31.08.2023	Magnolia, TX	merrick_bank	5	Very easy to use to view statements and make o...	NaN
1	Julicia	23.08.2023	Columbus, GA	merrick_bank	5	Merrick Bank has always been good to me for bu...	NaN
2	Karen	2.06.2023	Marrero, LA	merrick_bank	4	Times are tough for everyone and I have worked...	3.0
3	Brent	29.03.2023	Moultrie, GA	merrick_bank	5	I can not asked for a better Credit Card Compa...	3.0
4	Sharon	23.11.2022	Burnham, IL	merrick_bank	5	Updated on 02/10/2023: I was happy to sign for...	3.0
...	...	...	...	...	...	...	...
19266	J.	30.01.2017	Salem, OR	tcf_bank	1	Paid my 1st payment on time. They sent me a la...	11.0
19267	Destiny	28.01.2017	Andover, MN	tcf_bank	1	I have banked with TCF for about 4 years now a...	12.0
19268	Sean	25.01.2017	Bothell, WA	tcf_bank	1	Most inconvenient bank ever. As a business own...	10.0
19269	Edgar	12.01.2017	Minneapolis, MI	tcf_bank	1	Well I've been with TCF Bank for 3 plus years ...	12.0
19270	edward	2.01.2017	Suite B, MI	tcf_bank	1	Deposited $800 3 days ago by certified check a...	10.0

19271 rows × 7 columns

19271 Reviews

Data handling

pandas NumPy

Plotting

matplotlib

python + colab

scikit learn

PyTorch Transformers

Statistical modelling

Deep Learning implementation (BERT for this project)

1. Importing the libraries

Data handling & plotting the graph

Text cleaning for data preprocessing

Implementation the BERT model

Evaluation

```
# Import the necessary libraries for performing data analysis
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

# import necessary libraries for cleaning text
import nltk
nltk.download('stopwords')
nltk.download('punkt')
from nltk.corpus import stopwords
import re

# Install the required libraries for BERT
!pip install transformers
!pip install torch

# Import the necessary libraries for BERT
import torch
import torch.nn.functional as F
from transformers import BertTokenizer, BertForSequenceClassification, AdamW
from torch.utils.data import DataLoader, TensorDataset
from sklearn.model_selection import train_test_split
import time
from prettytable import PrettyTable

# Import the necessary libraries for evaluation
from sklearn.metrics import roc_curve, roc_auc_score, accuracy_score
```

2. Data preprocessing

	author	date	location	bank	star	text	like
0	Kyle	31.08.2023	Magnolia, TX	merrick_bank	5	Very easy to use to view statements and make o...	NaN
1	Julicia	23.08.2023	Columbus, GA	merrick_bank	5	Merrick Bank has always been good to me for bu...	NaN
2	Karen	2.06.2023	Marrero, LA	merrick_bank	4	Times are tough for everyone and I have worked...	3.0
3	Brent	29.03.2023	Moultrie, GA	merrick_bank	5	I can not asked for a better Credit Card Compa...	3.0
4	Sharon	23.11.2022	Burnham, IL	merrick_bank	5	Updated on 02/10/2023: I was happy to sign for...	3.0

- Define 4-5 stars as 1 (positive reviews), and 1-3 stars as 0 (Negative reviews) in ”Sentiment” column
- Remove punctuation, numbers, and stopwords
- Convert into lowercase

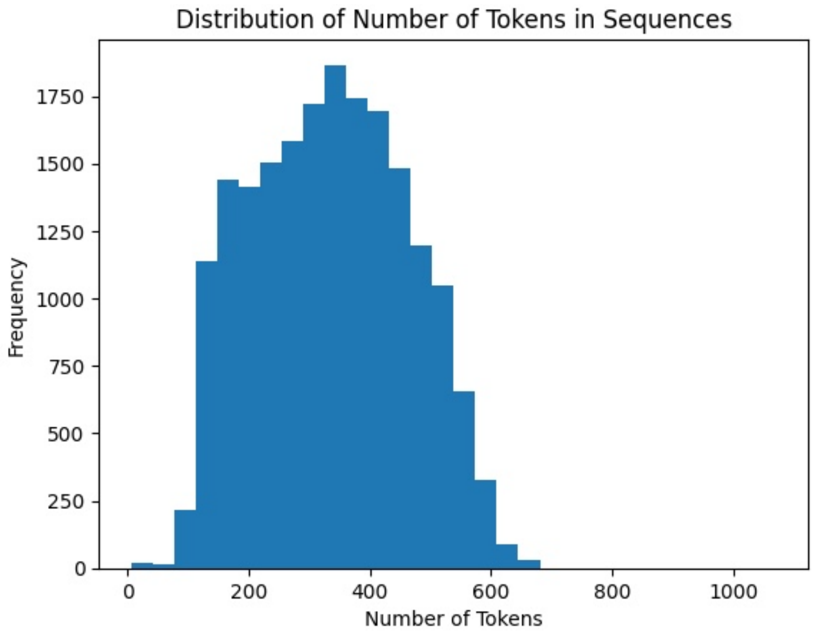
	star	text	sentiment
0	5	easy use view statements make online payments...	1
1	5	merrick bank always good business rely merric...	1
2	4	times tough everyone worked hard get credit r...	1
3	5	asked better credit card company merrick bank...	1
4	5	updated happy sign new credit card merrick co...	1

19181 Reviews (After cleaning)

2. Data preprocessing

Data statistics

	Sentiment	#Reviews	Min. #tokens	Avg. #tokens	Max. #tokens
0	Positive reviews	1869	8	212	643
1	Negative reviews	17312	6	349	1071



Most of the reviews have 200-400 tokens
Avg. for positive: 212
Avg. for negative: 349

I will define **the maximum sequence length for BERT implementation at 256**
(Not too short and too long)

3. Splitting dataset & Encoding the text

Inputs	Labels
text	**sentiment**
easy use view statements make online payments...	1
merrick bank always good business rely merric...	1
times tough everyone worked hard get credit r...	1
asked better credit card company merrick bank...	1
updated happy sign new credit card merrick co...	1

80%: Training dataset

10%: Validation dataset

10%: Test dataset

```
# Load pre-trained BERT tokenizer
tokenizer = BertTokenizer.from_pretrained('bert-base-uncased')
```

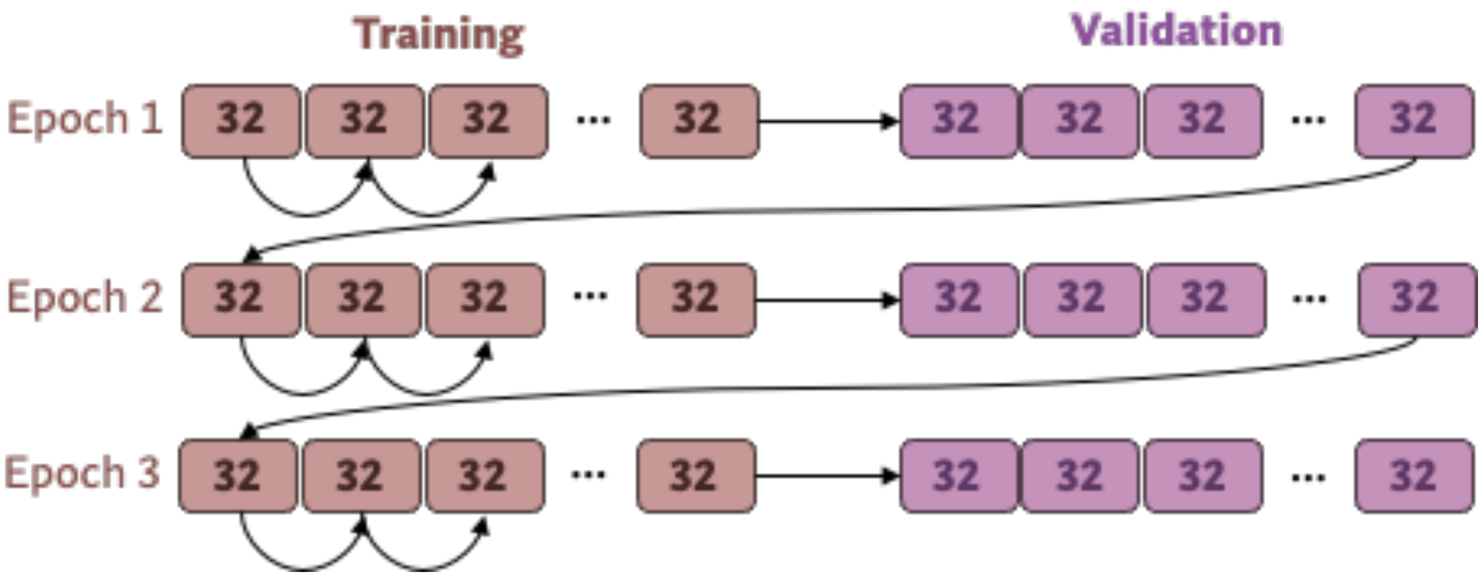
Encodings using pre-trained BERT tokenizer (Padding & Truncation with the maximum length at 256)

```
{'input_ids': tensor([[  101,  5223, 12978,  ...,      0,      0,      0],
        [  101,  2034,  5741,  ...,      0,      0,      0],
        [  101,  1052, 12273,  ...,      0,      0,      0],
        ...,
        [  101,  2288,  4003,  ...,      0,      0,      0],
        [  101,  4162,  3784,  ...,      0,      0,      0],
        [  101, 10263,  3361,  ...,      0,      0,      0]]), 'token_type_ids': tensor([[0, 0, 0,  ..., 0, 0, 0],
        [0, 0, 0,  ..., 0, 0, 0],
        [0, 0, 0,  ..., 0, 0, 0],
        ...,
        [0, 0, 0,  ..., 0, 0, 0],
        [0, 0, 0,  ..., 0, 0, 0],
        [0, 0, 0,  ..., 0, 0, 0]]), 'attention_mask': tensor([[1, 1, 1,  ..., 0, 0, 0],
        [1, 1, 1,  ..., 0, 0, 0],
        [1, 1, 1,  ..., 0, 0, 0],
        ...,
        [1, 1, 1,  ..., 0, 0, 0],
        [1, 1, 1,  ..., 0, 0, 0],
        [1, 1, 1,  ..., 0, 0, 0]])}
```

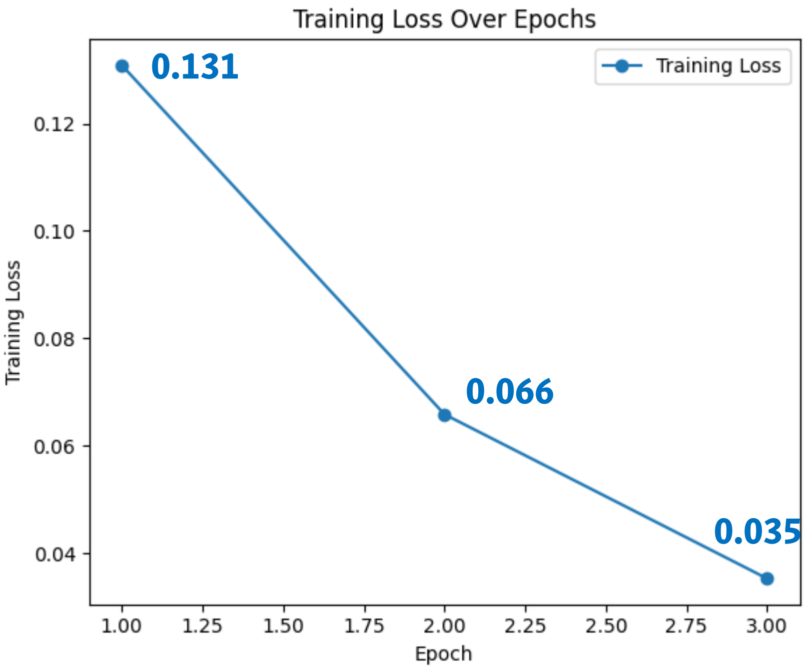
4. Model training

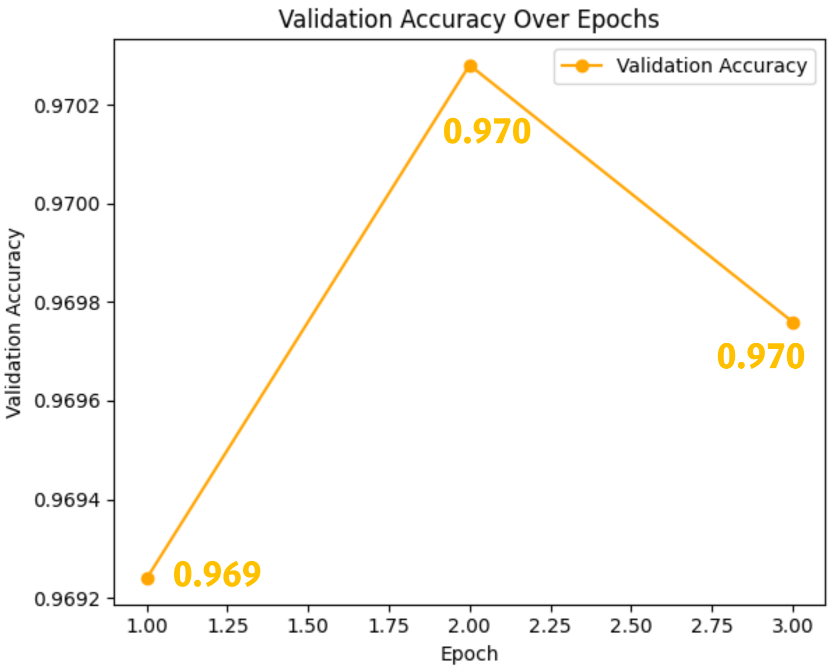
```
# Load Pre-trained BERT Model
model = BertForSequenceClassification.from_pretrained('bert-base-uncased', num_labels=2)
```

Batch size: 32, #Epochs: 3, Optimizer: AdamW with initial learning rate at 2e-5



4. Model training

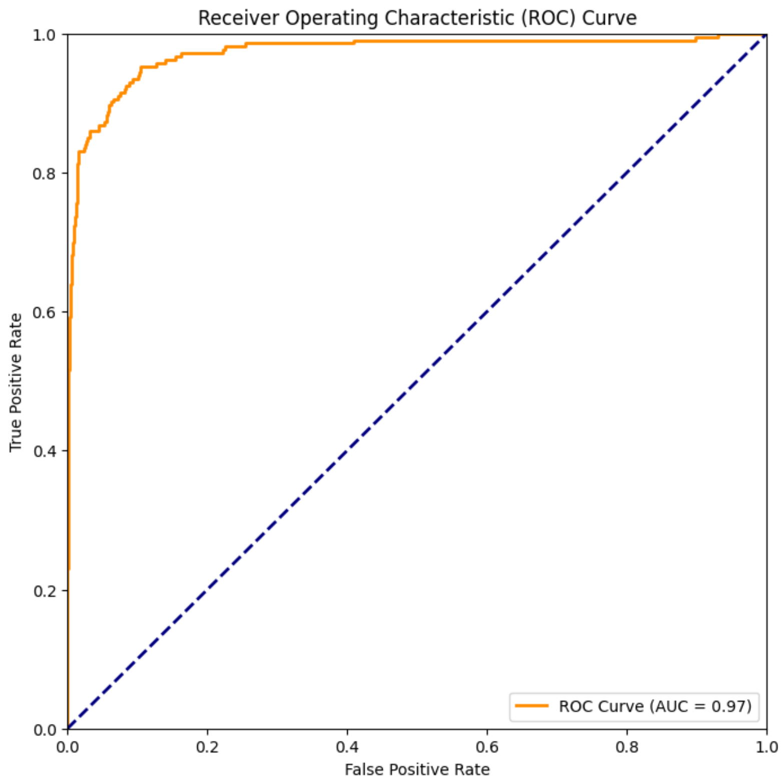




Epoch	Training Loss	Validation Accuracy	Training Time (s)	Max GPU Memory Allocated (MB)
1	0.1308751515694894	0.9692387904066736	419.7340638637543	5733.5380859375
2	0.06584998913725333	0.970281543274244	422.6290991306305	5733.5390625
3	0.035265509981157565	0.9697601668404588	422.54603147506714	5733.5390625

5. Evaluation result

Evaluate the model performance to classify the reviews between positive and negative reviews on Test dataset



`ROC-AUC Score: 0.9730`

The model is **excellent at telling the difference between positive and negative reviews**

`Accuracy: 0.9656`

The overall **correctness of the model's predictions is excellent**

5. Evaluation result

Evaluate the model performance to classify the reviews between positive and negative reviews on Test dataset

Manually check with 10 unseen reviews

```
# 10 unseen reviews for manual test
unseen_reviews = ["Impressed with the bank's online services, making transactions has never been easier.",
                  "Customer support was unhelpful and frustrating; it took forever to resolve a simple issue.",
                  "Received a great interest rate on my savings account, very satisfied with the banking experience.",
                  "The mobile app is user-friendly and convenient for managing accounts on the go.",
                  "Unexpected fees and hidden charges made my experience with this bank disappointing.",
                  "Quick and efficient loan approval process, would recommend for financial assistance.",
                  "The staff at the local branch were friendly and assisted me with professionalism.",
                  "I've been a customer for years, and the bank has consistently met my financial needs.",
                  "The credit card application process was straightforward, and I got approved quickly.",
                  "Poor security measures; my account was compromised, and the recovery process was frustrating."]

# Sentiment for 10 unseen reviews
sentiment_unseen_reviews = ['Positive', 'Negative', 'Positive', 'Positive', 'Negative', 'Positive', 'Positive', 'Positive', 'Positive', 'Negative']

# Clean the text
cleaned_unseen_reviews = []
for text in unseen_reviews:
    cleaned_text = clean_text(text)
    cleaned_unseen_reviews.append(cleaned_text)

# Tokenize and encode the unseen reviews
unseen_encodings = tokenizer(unseen_reviews, padding=True, truncation=True, return_tensors='pt')

# Make predictions
model.eval()
with torch.no_grad():
    inputs = {key: unseen_encodings[key].to(model.device) for key in unseen_encodings}
    outputs = model(**inputs)
    logits = outputs.logits
    predictions = torch.argmax(logits, dim=1)
```

5. Evaluation result

All correct classification

```
Review: Impressed with the bank's online services, making transactions has never been easier.
Predicted Sentiment: Positive
True Sentiment: Positive
```

Positive

```
Review: Customer support was unhelpful and frustrating; it took forever to resolve a simple issue.
Predicted Sentiment: Negative
True Sentiment: Negative
```

Negative

```
Review: Received a great interest rate on my savings account, very satisfied with the banking experience.
Predicted Sentiment: Positive
True Sentiment: Positive
```

Positive

```
Review: The mobile app is user-friendly and convenient for managing accounts on the go.
Predicted Sentiment: Positive
True Sentiment: Positive
```

Positive

```
Review: Unexpected fees and hidden charges made my experience with this bank disappointing.
Predicted Sentiment: Negative
True Sentiment: Negative
```

Negative

```
Review: Quick and efficient loan approval process, would recommend for financial assistance.
Predicted Sentiment: Positive
True Sentiment: Positive
```

Positive

```
Review: The staff at the local branch were friendly and assisted me with professionalism.
Predicted Sentiment: Positive
True Sentiment: Positive
```

Positive

```
Review: I've been a customer for years, and the bank has consistently met my financial needs.
Predicted Sentiment: Positive
True Sentiment: Positive
```

Positive

```
Review: The credit card application process was straightforward, and I got approved quickly.
Predicted Sentiment: Positive
True Sentiment: Positive
```

Positive

```
Review: Poor security measures; my account was compromised, and the recovery process was frustrating.
Predicted Sentiment: Negative
True Sentiment: Negative
```

Negative

Conclusion

In this sentiment analysis of bank customer reviews using BERT, **the model achieved a high ROC score of 0.973**, indicating excellent ability to distinguish between positive and negative sentiments. **The accuracy of 96.56%** reflects the model's overall correctness in predicting sentiment, showcasing its **strong performance in classifying customer reviews**.

With **the training time at around 420 seconds and 5,500 MB memory usage for each epoch** for a dataset with more than 15,000 rows seem reasonable this BERT model.

Thank you

for your attention