Project MS2

Weihua Pan
IE7374 -- Gen AI
4/7/2024

Introduction

As Large Language Models (LLMs) continue to evolve, they are characterized by increasingly larger parameters. However, the pace of computational resource development struggles to keep up. Consequently, there is a significant advantage in devising a framework that can reduce parameter size while preserving performance. Such a framework would enable these models to operate on devices with limited computational capabilities, including mobile phones.

Problem Statement

This project is aimed to develop a car chatbot that will give some comment on specific car model and probably give a recommend car by some budget and features.
In addition, this project is focused on discovering methods to decrease model size without compromising performance for specified NLP tasks. I will identify a Large Language Model (LLM) like GPT-4 or BERT that is appropriate for this project and subsequently develop a user interface tailored for real time text analysis.

Dataset

The dataset i am using is scrape by `ankkur13`. The dataset contains reviews and rating for each car brands, model and etc. Here is the link: https://github.com/ankkur13/Edmunds-Car-Consumer-Ratings-and-Reviews

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

In [ ]:
```
df = pd.read_csv('Scraped Data/Scraped_Car_Review_eagle.csv',lineterminator='\n',in
```

In [ ]:
```
df
```

Out[ ]:

	Review_Date	Author_Name	Vehicle_Title	Review_Title	Review	Rating\r
0	on 12/14/05 20:47 PM (PST)	Trisha	1997 Eagle Talon Hatchback ESi 2dr Hatchback	AWESOME CAR!	This is a fantastic car! Not only is it fun t...	5.000
1	on 12/04/05 00:04 AM (PST)	Patricia	1997 Eagle Talon Hatchback ESi 2dr Hatchback	Awesome Car!	This car is the most fun car I have ever driv...	5.000
2	on 02/19/05 14:20 PM (PST)	Paco	1997 Eagle Talon Hatchback 2dr Hatchback	Not Bad.	Most people don't like these cars, think\rthe...	4.250
3	on 08/06/04 00:00 AM (PDT)	strave	1997 Eagle Talon Hatchback ESi 2dr Hatchback	enjoyable	for a four cylander with no turbo it is \ran ...	4.875
4	on 12/23/03 00:00 AM (PST)	Deelio	1997 Eagle Talon Hatchback ESi 2dr Hatchback	Fun, Reliable Car	This was my first brand spankin' new \rcar an...	4.750
5	on 12/02/03 00:00 AM (PST)	D. Strickland	1997 Eagle Talon Hatchback ESi 2dr Hatchback	ESi	Car is fun to drive, most DSM vehicles \rhave...	4.625
6	on 05/07/03 00:00 AM (PDT)	DR1665	1997 Eagle Talon Hatchback 2dr Hatchback	pocket rocket!	I have owned my Talon since it was \rnew. Th...	5.000
7	on 09/26/02 00:00 AM (PDT)	Andy Ashcraft	1997 Eagle Talon Hatchback ESi 2dr Hatchback	I love mine!	I've had nothing but fun driving my '97 \rTal...	4.625
8	on 09/02/02 00:00 AM (PDT)	Mitch'sTalon	1997 Eagle Talon Hatchback ESi 2dr Hatchback	One of the funnest cars to drive!	If you don't own one or have never \rdriven o...	4.750
9	on 09/07/14 00:22 AM	mack_kenamond	1997 Eagle Talon TSi TSi	18 years of fun	In case you didn't know, this car is	4.875

	Review_Date	Author_Name	Vehicle_Title	Review_Title	Review	Rating\r
	(PDT)		Turbo 2dr Hatchback AWD		virtuall...	
10	on 01/29/10 07:42 AM (PST)	Mikek	1997 Eagle Talon TSi TSi Turbo 2dr Hatchback AWD	Have to mod it	97 talon tsi awd, Swapped a 1G (91) 6 bolt mo...	5.000
11	on 11/24/06 10:18 AM (PST)	DarkPhoenix	1997 Eagle Talon TSi TSi Turbo 2dr Hatchback AWD	AWD Bliss	Being a previous owner of DSMs, this is by fa...	4.625
12	on 06/12/05 17:43 PM (PDT)	god	1997 Eagle Talon TSi TSi Turbo 2dr Hatchback AWD	Talon tsi	Great car fun to drive fast as anything. Easy...	4.125
13	on 02/19/05 12:03 PM (PST)	Stewart Gibb	1997 Eagle Talon TSi TSi Turbo 2dr Hatchback AWD	Best Car I've Owned	I bought it after one drive because of \rthe ...	5.000
14	on 12/14/04 22:58 PM (PST)	RjH	1997 Eagle Talon TSi TSi Turbo 2dr Hatchback AWD	great car	This is an amazing car. Mine\rcrankwalked on...	4.125
15	on 04/16/04 00:00 AM (PDT)	Richard M.	1997 Eagle Talon TSi TSi Turbo 2dr Hatchback AWD	Excellent Performer/ minor flaws	Performance is great as is aftermarket \rsupp...	5.000
16	on 09/28/03 00:00 AM (PDT)	Talia	1997 Eagle Talon TSi TSi Turbo 2dr Hatchback	LOVE this car!!! :)	This car can kick it! I have the AWD \rturbo ...	5.000
17	on 02/28/03 00:00 AM (PST)	Sethsoccer	1997 Eagle Talon TSi TSi Turbo 2dr Hatchback AWD	Get Insurance That Covers Towing	Fun car, not the fasted though. Off the \rlin...	3.250
18	on 02/17/03 00:00 AM (PST)	irrelevant	1997 Eagle Talon TSi TSi Turbo 2dr Hatchback	Get the turbo.	This car will rock your face off. The \rmost...	5.000

	Review_Date	Author_Name	Vehicle_Title	Review_Title	Review	Rating\r
19	on 10/17/02 00:00 AM (PDT)	james ossey	1997 Eagle Talon TSi TSi Turbo 2dr Hatchback AWD	the best	all, this car is the best car next to \rthe ...	5.000
20	on 10/14/02 00:00 AM (PDT)	Mantas	1997 Eagle Talon TSi TSi Turbo 2dr Hatchback AWD	97 Talon Tsi Turbo	Ok Negatives first. Because its an AWD \rit ...	4.750
21	on 08/08/02 00:00 AM (PDT)	COOL	1997 Eagle Talon TSi TSi Turbo 2dr Hatchback	this is the worst day of my life	Let me tell you a little story about the eagl...	1.000
22	on 05/03/15 20:05 PM (PDT)	zombiemotors	1998 Eagle Talon Hatchback ESi 2dr Hatchback	Problems of age	Well I was given a 1998 Eagle Talon as a gift...	3.375
23	on 10/28/09 07:30 AM (PDT)	johnie tyler	1998 Eagle Talon Hatchback ESi 2dr Hatchback	1998 eagle talon	i bought my 1998 talon esi almost two months ...	5.000
24	on 09/10/08 20:03 PM (PDT)	avril89lavigne	1998 Eagle Talon Hatchback ESi 2dr Hatchback	1998 Eagle Talon ESi	The car so far has done well. One common prob...	4.625
25	on 02/22/03 00:00 AM (PST)	Ulysses S. Grant	1998 Eagle Talon Hatchback ESi 2dr Hatchback	IIIIIIIII love it!	I love it! I love it! I love it!! \rGreat rid...	4.875
26	on 10/24/02 00:00 AM (PDT)	James Zalin	1998 Eagle Talon Hatchback ESi 2dr Hatchback	Eagle talon 98	My eagle talon is one of the best cars \rI've...	5.000
27	on 07/03/02 00:00 AM (PDT)	Sandy Dunbar	1998 Eagle Talon Hatchback ESi 2dr Hatchback	Good	The car is fine. However, the back \rspindle ...	4.625
28	on 11/26/08 11:00 AM (PST)	Bob's Garage	1998 Eagle Talon TSi TSi Turbo 2dr	Belt Problems	Got this car about a year and a half ago and ...	4.375

	Review_Date	Author_Name	Vehicle_Title	Review_Title	Review	Rating\r
			Hatchback AWD			
29	on 10/31/08 03:16 AM (PDT)	Al Simmons	1998 Eagle Talon TSi TSi Turbo 2dr Hatchback	The Ultimate Eagle Talon - 1998 Model	I owned a 1992 Eagle Talon from '92-97 and ha...	4.875
30	on 04/21/03 00:00 AM (PDT)	Tammara	1998 Eagle Talon TSi TSi Turbo 2dr Hatchback	Encredible!	Talons are the greatest! They are the \rmost ...	5.000
31	on 03/28/03 00:00 AM (PST)	Kezer	1998 Eagle Talon TSi TSi Turbo 2dr Hatchback AWD	Best Car Ever Owned!!!!	I bought this car with 72 000km on it. I\rbou...	5.000
32	on 01/08/03 00:00 AM (PST)	eagletalonTSI	1998 Eagle Talon TSi TSi Turbo 2dr Hatchback AWD	Eagle Talon TS AWD Hatchback	This car is great, theirs many mods for \rit ...	5.000
33	on 12/26/02 00:00 AM (PST)	watercrit	1998 Eagle Talon TSi TSi Turbo 2dr Hatchback AWD	Factory can't fix	Spitzer dodge (Lakewood), the dealer \rwho Ch...	4.125
34	on 09/24/02 00:00 AM (PDT)	DEEB0rider	1998 Eagle Talon TSi TSi Turbo 2dr Hatchback AWD	143,000 miles in 4 years and like new	I purchased this car in february of \r2002. ...	5.000
35	on 07/19/02 00:00 AM (PDT)	David Reid	1998 Eagle Talon TSi TSi Turbo 2dr Hatchback	Best Car Ever Owned!!!!!	My talon is an amazing car, ive owned it\rsin...	5.000
36	on 07/27/13 16:11 PM (PDT)	n5061m	1997 Eagle Vision Sedan TSi 4dr Sedan	a great car, beats ever new model of other makes	easily one of the best cars i ever owned.wish...	5.000
37	on 10/31/09 05:50 AM (PDT)	rgfrit	1997 Eagle Vision Sedan TSi 4dr Sedan	BEST CAR EVER MADE IN THE USA	I can say that these cars are Fantastic! The...	5.000
38	on 09/01/06 19:57 PM (PDT)	brian	1997 Eagle Vision Sedan	Better than a Timex	I bought this car when it had 83,000 miles on...	4.750

	Review_Date	Author_Name	Vehicle_Title	Review_Title	Review	Rating\r
			ESi 4dr Sedan			
39	on 02/22/04 00:00 AM (PST)	Decoy72	1997 Eagle Vision Sedan TSi 4dr Sedan	If you can find one... buy it!	This car is great car in almost every \rarea ...	4.375
40	on 01/26/04 00:00 AM (PST)	hockhog203	1997 Eagle Vision Sedan ESi 4dr Sedan	70K and Bullet proof	Bought this car as a leftover in 1998 \rfor $...	4.625
41	on 01/11/04 00:00 AM (PST)	thiel433	1997 Eagle Vision Sedan TSi 4dr Sedan	Great Full Size Sedan	This full size sedan has been nothing \rshort...	5.000
42	on 12/31/03 00:00 AM (PST)	manning_mike	1997 Eagle Vision Sedan TSi 4dr Sedan	Have loved this car	I have had this car for over five years\rnow....	4.875
43	on 02/07/03 00:00 AM (PST)	Mike G	1997 Eagle Vision Sedan ESi 4dr Sedan	Hi Performance Sedan	I've always liked sedans for their \rride and...	5.000
44	on 08/31/02 00:00 AM (PDT)	Bob M	1997 Eagle Vision Sedan TSi 4dr Sedan	Good Fast Car	This is easly one of the best cars I've \rown...	4.875

Let's take a look at the structure in one csv file. The columns are `Review_Date, Author_Name, Vehicle_Title, Review_Title, Review, Rating\r`. I don't think having date and author_name would be helpful to train the car Chatbot. So, I will ignore them and transform the csv to txt like this.

Vehicle: 1997 Eagle Talon Hatchback ESi 2dr Hatchback
Title: AWESOME CAR!
Review: "This is a fantastic car! Not only is it fun to drive, but it is a head turner. I have had more compliments on this car than I have had on any other car I have owned. I haven't had any major repairs on this car. It has been very dependable. I love driving it. It is red with the black roof & spoiler, which is very sharp. I haven't seen very many on the road, & when I take it in for oil changes, the mechanics tell me they have never seen one in such good condition. I will keep this car until it just won't go anymore. I now have 55,000 miles on it & expect to put a lot more miles on it before it dies. I was disappointed to hear that 1998 was the last year for this car."
Rating: 5.0 [round to 1 decimal]

And then next car review

In [ ]:
```
# put all cvs filename to one list
cvs_list = []
with open("Scraped Data/cvs_list.txt",mode="r",encoding='utf-8') as f:
    cvs_list = [line.strip() for line in f.readlines()]
```

In [ ]:
```
def df_transform_txt(df: pd.DataFrame) -> str:
    """Extract  text and format them from csv dataframe.
    Only transform 1% of the data
    """
    df.columns = df.columns.str.strip()
    txt_lines = []
    one_percent = 0.01 * len(df)
    for _, row in df.iterrows():
        if _ >= one_percent:
            break
        vehicle = row['Vehicle_Title']
        title = row['Review_Title']
        review = row['Review'].replace("\r",' ') # fix the review format
        rating = round(row["Rating"], 1)

        txt_lines.append(f"Vehicle: {vehicle}\nTitle: {title}\nReview: {review}\nRa

    # Join all the lines into a single string with two newlines separating each ent
    txt = "\n\n".join(txt_lines)
    return txt
```

In [ ]:
```
# loop through each csv file and transform them into 1 txt file
car_reviews_text_list = []
for filename in cvs_list:
    print(filename)
    df = pd.read_csv(f"Scraped Data/{filename}",lineterminator="\n",index_col=0)
    text = df_transform_txt(df)
    car_reviews_text_list.append(text)

car_reviews_text = "\n".join(car_reviews_text_list)
```

```
Scraped_Car_Review_daewoo.csv
Scraped_Car_Review_dodge.csv
Scraped_Car_Review_eagle.csv
Scraped_Car_Review_ferrari.csv
Scraped_Car_Review_fiat.csv
Scraped_Car_Review_fisker.csv
Scraped_Car_Review_ford.csv
Scraped_Car_Review_genesis.csv
Scraped_Car_Review_geo.csv
Scraped_Car_Review_hummer.csv
Scraped_Car_Review_hyundai.csv
Scraped_Car_Review_infiniti.csv
Scraped_Car_Review_isuzu.csv
Scraped_Car_Review_jaguar.csv
Scraped_Car_Review_jeep.csv
Scraped_Car_Review_kia.csv
Scraped_Car_Review_lamborghini.csv
Scraped_Car_Review_land-rover.csv
Scraped_Car_Review_lexus.csv
Scraped_Car_Review_lincoln.csv
Scraped_Car_Review_lotus.csv
Scraped_Car_Review_maserati.csv
Scraped_Car_Review_maybach.csv
Scraped_Car_Review_mazda.csv
Scraped_Car_Review_mclaren.csv
Scraped_Car_Review_mercedes-benz.csv
Scraped_Car_Review_mercury.csv
Scraped_Car_Review_mini.csv
Scraped_Car_Review_mitsubishi.csv
Scraped_Car_Review_nissan.csv
Scraped_Car_Review_oldsmobile.csv
Scraped_Car_Review_panoz.csv
Scraped_Car_Review_plymouth.csv
Scraped_Car_Review_pontiac.csv
Scraped_Car_Review_porsche.csv
Scraped_Car_Review_ram.csv
Scraped_Car_Review_rolls-royce.csv
Scraped_Car_Review_saab.csv
Scraped_Car_Review_saturn.csv
Scraped_Car_Review_scion.csv
Scraped_Car_Review_smart.csv
Scraped_Car_Review_spyker.csv
Scraped_Car_Review_subaru.csv
Scraped_Car_Review_suzuki.csv
Scraped_Car_Review_tesla.csv
Scraped_Car_Review_volvo.csv
Scrapped_Car_Review_Bugatti.csv
Scrapped_Car_Review_Buick.csv
Scrapped_Car_Review_Cadillac.csv
Scrapped_Car_Review_Chevrolet.csv
Scrapped_Car_Review_chrysler.csv
Scrapped_Car_Reviews_Acura.csv
Scrapped_Car_Reviews_AlfaRomeo.csv
Scrapped_Car_Reviews_AMGeneral.csv
Scrapped_Car_Reviews_AstonMartin.csv
Scrapped_Car_Reviews_Audi.csv
```

```
Scrapped_Car_Reviews_Bentley.csv
Scrapped_Car_Reviews_BMW.csv
Scrapped_Car_Reviews_GMC.csv
Scrapped_Car_Reviews_Honda.csv
Scrapped_Car_Reviews_Toyota.csv
Scrapped_Car_Reviews_Volkswagen.csv
```

In [ ]:
```python
# define output file path
output_file_path = 'car_reviews.txt'

# write to the file
with open(output_file_path, 'w', encoding='utf-8') as file:
    file.write(car_reviews_text)

print(f"Review text has been saved to '{output_file_path}'")
```

```
Review text has been saved to 'car_reviews.txt'
```

Modeling

In [ ]:
```python
import torch
from torch import nn
from torch.utils.data import Dataset, DataLoader
from torchtext.data.utils import get_tokenizer
from torchtext.vocab import build_vocab_from_iterator
from torch.optim import Adam
```

In [ ]:
```python
device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
```

In [ ]:
```python
device
```

Out[ ]:
```
device(type='cuda')
```

In [ ]:
```python
# Lowercase the text
text = car_reviews_text.lower()

# Define the tokenizer
tokenizer = get_tokenizer('basic_english')

# Tokenize the text
tokenized_text = [list(tokenizer(text))]

# Build the vocabulary from the tokenized text
vocab = build_vocab_from_iterator(tokenized_text)

# Numericalize the text
numericalized_text = [vocab[token] for token in tokenized_text[0]]
```

In [ ]:
```python
token_list = numericalized_text[0:10]
```

In [ ]:
```python
vocab.lookup_tokens(token_list)
```

Out[ ]:
```
['vehicle',
 '2000',
 'daewoo',
 'leganza',
 'sedan',
 'sx',
 '4dr',
 'sedan',
 'title',
 'best']
```

see if the numericalized_text is reversible

In [ ]:
```
len(vocab)
```

Out[ ]:
```
12444
```

the vocab size is 12444 which is quite large for 1mb text file.

In [ ]:
```
# Define the dataset
class LlamaDataset(Dataset):
    def __init__(self, text, sequence_length):
        self.text = text
        self.sequence_length = sequence_length

    def __len__(self):
        return len(self.text) - self.sequence_length

    def __getitem__(self, idx):
        return (
            torch.tensor(self.text[idx:idx+self.sequence_length]),
            torch.tensor(self.text[idx+1:idx+self.sequence_length+1]),
        )

# Create the dataset and dataloader
sequence_length = 32
dataset = LlamaDataset(numericalized_text, sequence_length)
dataloader = DataLoader(dataset, batch_size=128)
```

In [ ]:
```
class LlamaModel(nn.Module):
    def __init__(self, vocab_size, embed_size, hidden_size, num_layers, num_heads,
        super().__init__()
        self.embedding = nn.Embedding(vocab_size, embed_size)
        self.transformer = nn.Transformer(
            d_model=embed_size,
            nhead=num_heads,
            num_encoder_layers=num_layers,
            num_decoder_layers=num_layers,
            dim_feedforward=hidden_size,
            dropout=dropout,
            batch_first=True,
        )
        self.fc = nn.Linear(embed_size, vocab_size)
```

```
    def forward(self, x):
        embedded = self.embedding(x)
        output = self.transformer(embedded, embedded)
        output = self.fc(output)
        return output
```

In [ ]:

```
# Initialize the model and the optimizer
model = LlamaModel(len(vocab), embed_size=128, hidden_size=128, num_layers=2, num_h

# If there are multiple GPUs, wrap the model with nn.DataParallel
if torch.cuda.device_count() > 1:
    print("Let's use", torch.cuda.device_count(), "GPUs!")
    model = nn.DataParallel(model)
model = model.to(device)

optimizer = Adam(model.parameters(), lr=0.001)

# Train the model
for epoch in range(20):
    for batch in dataloader:
        x, y = batch
        x = x.to(device)
        y = y.to(device)
        optimizer.zero_grad()
        y_pred = model(x)
        loss = nn.functional.cross_entropy(y_pred.view(-1, len(vocab)), y.view(-1))
        loss.backward()
        optimizer.step()
    print(f'Epoch {epoch}, Loss {loss.item()}')
    if float(loss.item()) < 0.06:
        break
```

```
c:\Users\panwe\Desktop\IE7374_Project2\LLM\Lib\site-packages\torch\nn\functional.py:
5476: UserWarning: 1Torch was not compiled with flash attention. (Triggered internal
ly at ..\aten\src\ATen\native\transformers\cuda\sdp_utils.cpp:263.)
  attn_output = scaled_dot_product_attention(q, k, v, attn_mask, dropout_p, is_causa
l)
```

```
Epoch 0, Loss 5.0411696434021
Epoch 1, Loss 4.13360071182251
Epoch 2, Loss 3.409588098526001
Epoch 3, Loss 2.9343554973602295
Epoch 4, Loss 2.6019606590270996
Epoch 5, Loss 2.356454849243164
Epoch 6, Loss 2.1351141929626465
Epoch 7, Loss 2.0737743377685547
Epoch 8, Loss 1.949976921081543
Epoch 9, Loss 1.8891923427581787
Epoch 10, Loss 1.7727535963058472
Epoch 11, Loss 1.6936289072036743
Epoch 12, Loss 1.6191486120224
Epoch 13, Loss 1.5524119138717651
Epoch 14, Loss 1.4959806203842163
Epoch 15, Loss 1.4818544387817383
Epoch 16, Loss 1.3750232458114624
Epoch 17, Loss 1.4040517807006836
Epoch 18, Loss 1.2730814218521118
Epoch 19, Loss 1.3316398859024048
```

The model takes 1min to run 1 epoch. For the first 20 epoch, the loss function is lower to 1.3, and the model is converging.

In [ ]:
```python
# Use the trained model to generate new text
def generate_text(model, human_input, num_tokens):
    model.eval()  # Set the model to evaluation mode
    with torch.no_grad():  # No need to track the gradients
        tokens = [vocab[token] for token in tokenizer(human_input)]
        tokens = torch.tensor(tokens).unsqueeze(0).to(device)
        for _ in range(num_tokens):
            output = model(tokens)
            probabilities = nn.functional.softmax(output[0, -1], dim=0)
            next_token = torch.multinomial(probabilities, 1).item()
            tokens = torch.cat([tokens, torch.tensor([[next_token]]).to(device)], d
        generated_text = ' '.join(vocab.get_itos()[token] for token in tokens[0].cp
        return generated_text
```

Experiment Findings

In [ ]:
```python
result = generate_text(model, human_input="I love tesla  ", num_tokens=32)
print(result)
```

```
i love tesla love rating a performance a performance performance performance perform
ance performance performance performance performance performance performance perform
ance performance performance performance performance performance performance perform
ance performance performance performance performance performance performance perform
ance performance performance
```

from the generated output above, we can see tesla have some relation to performance. This is interesting, tesla indeed have some tags of performance car.

In [ ]:
```
result = generate_text(model, human_input="vehicle 2012 bmw m3", num_tokens=32)
print(result)
```

```
vehicle 2012 bmw m3 m3 m3 m3 m3 m3 m3 m3 m3 m3 m3 m3 m3 m3 m3 m3 m3 m3 all cost all
cost all cost all cost all cost all cost cost cost cost
```

with m3 as input, it output some cost tokens in the end.

Conclusion

The model can output some words, but in general, the sentences is not making sense. I will see whether increasing epochs can improve the performance. Another way to improve is increasing the data size, but it will dramatically increase the computation and I don't think it is trainable using one GPU. I think the best way would be getting ultra small LLM like llamma 7b to fine-tuning using the dataset.