cs541-hw1-final-2

February 26, 2024

1 CS 541-A-Homework 1

1.1 K nearest neighbors and distance metrics

1.1.1 *Fill your details below*

1.1.2 Name: Sneha Venkatesh

1.1.3 CWID: 20027527

1.1.4 Email ID: svenkate1@stevens.edu

1.1.5 References: *Cite your references here*

1.1.6 Install prerequisites

[1]:
```
%pip install sentencepiece datasets
%pip install git+https://github.com/huggingface/transformers
```

```
Requirement already satisfied: sentencepiece in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(0.2.0)
Requirement already satisfied: datasets in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(2.17.0)
Requirement already satisfied: filelock in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from datasets) (3.13.1)
Requirement already satisfied: numpy>=1.17 in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from datasets) (1.26.4)
Requirement already satisfied: pyarrow>=12.0.0 in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from datasets) (15.0.0)
Requirement already satisfied: pyarrow-hotfix in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from datasets) (0.6)
Requirement already satisfied: dill<0.3.9,>=0.3.0 in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
```

```
(from datasets) (0.3.8)
Requirement already satisfied: pandas in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from datasets) (2.2.0)
Requirement already satisfied: requests>=2.19.0 in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from datasets) (2.31.0)
Requirement already satisfied: tqdm>=4.62.1 in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from datasets) (4.66.2)
Requirement already satisfied: xxhash in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from datasets) (3.4.1)
Requirement already satisfied: multiprocess in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from datasets) (0.70.16)
Requirement already satisfied: fsspec<=2023.10.0,>=2023.1.0 in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from fsspec[http]<=2023.10.0,>=2023.1.0->datasets) (2023.10.0)
Requirement already satisfied: aiohttp in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from datasets) (3.9.3)
Requirement already satisfied: huggingface-hub>=0.19.4 in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from datasets) (0.20.3)
Requirement already satisfied: packaging in
/Users/snehavenkatesh/Library/Python/3.12/lib/python/site-packages (from
datasets) (23.2)
Requirement already satisfied: pyyaml>=5.1 in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from datasets) (6.0.1)
Requirement already satisfied: aiosignal>=1.1.2 in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from aiohttp->datasets) (1.3.1)
Requirement already satisfied: attrs>=17.3.0 in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from aiohttp->datasets) (23.2.0)
Requirement already satisfied: frozenlist>=1.1.1 in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from aiohttp->datasets) (1.4.1)
Requirement already satisfied: multidict<7.0,>=4.5 in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from aiohttp->datasets) (6.0.5)
Requirement already satisfied: yarl<2.0,>=1.0 in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from aiohttp->datasets) (1.9.4)
Requirement already satisfied: typing-extensions>=3.7.4.3 in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
```

```
(from huggingface-hub>=0.19.4->datasets) (4.9.0)
Requirement already satisfied: charset-normalizer<4,>=2 in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from requests>=2.19.0->datasets) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from requests>=2.19.0->datasets) (3.6)
Requirement already satisfied: urllib3<3,>=1.21.1 in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from requests>=2.19.0->datasets) (2.2.1)
Requirement already satisfied: certifi>=2017.4.17 in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from requests>=2.19.0->datasets) (2024.2.2)
Requirement already satisfied: python-dateutil>=2.8.2 in
/Users/snehavenkatesh/Library/Python/3.12/lib/python/site-packages (from
pandas->datasets) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from pandas->datasets) (2024.1)
Requirement already satisfied: tzdata>=2022.7 in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from pandas->datasets) (2024.1)
Requirement already satisfied: six>=1.5 in
/Users/snehavenkatesh/Library/Python/3.12/lib/python/site-packages (from python-
dateutil>=2.8.2->pandas->datasets) (1.16.0)
Note: you may need to restart the kernel to use updated packages.
Collecting git+https://github.com/huggingface/transformers
  Cloning https://github.com/huggingface/transformers to
/private/var/folders/7h/8pn4zhvx5sz0kr6hl7fzpq_r0000gn/T/pip-req-build-aky7cjt3
  Running command git clone --filter=blob:none --quiet
https://github.com/huggingface/transformers
/private/var/folders/7h/8pn4zhvx5sz0kr6hl7fzpq_r0000gn/T/pip-req-build-aky7cjt3
  Resolved https://github.com/huggingface/transformers to commit
c8d98405a8f7b0e5d07391b671dcc61bb9d7bad5
  Installing build dependencies … done
  Getting requirements to build wheel … done
  Installing backend dependencies … done
  Preparing metadata (pyproject.toml) … done
Requirement already satisfied: filelock in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from transformers==4.39.0.dev0) (3.13.1)
Requirement already satisfied: huggingface-hub<1.0,>=0.19.3 in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from transformers==4.39.0.dev0) (0.20.3)
Requirement already satisfied: numpy>=1.17 in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from transformers==4.39.0.dev0) (1.26.4)
Requirement already satisfied: packaging>=20.0 in
```

```
/Users/snehavenkatesh/Library/Python/3.12/lib/python/site-packages (from
transformers==4.39.0.dev0) (23.2)
Requirement already satisfied: pyyaml>=5.1 in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from transformers==4.39.0.dev0) (6.0.1)
Requirement already satisfied: regex!=2019.12.17 in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from transformers==4.39.0.dev0) (2023.12.25)
Requirement already satisfied: requests in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from transformers==4.39.0.dev0) (2.31.0)
Requirement already satisfied: tokenizers<0.19,>=0.14 in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from transformers==4.39.0.dev0) (0.15.2)
Requirement already satisfied: safetensors>=0.4.1 in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from transformers==4.39.0.dev0) (0.4.2)
Requirement already satisfied: tqdm>=4.27 in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from transformers==4.39.0.dev0) (4.66.2)
Requirement already satisfied: fsspec>=2023.5.0 in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from huggingface-hub<1.0,>=0.19.3->transformers==4.39.0.dev0) (2023.10.0)
Requirement already satisfied: typing-extensions>=3.7.4.3 in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from huggingface-hub<1.0,>=0.19.3->transformers==4.39.0.dev0) (4.9.0)
Requirement already satisfied: charset-normalizer<4,>=2 in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from requests->transformers==4.39.0.dev0) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from requests->transformers==4.39.0.dev0) (3.6)
Requirement already satisfied: urllib3<3,>=1.21.1 in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from requests->transformers==4.39.0.dev0) (2.2.1)
Requirement already satisfied: certifi>=2017.4.17 in
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-packages
(from requests->transformers==4.39.0.dev0) (2024.2.2)
Note: you may need to restart the kernel to use updated packages.
```

1.1.7 Import relevant libraries

[2]:
```
from datasets import load_dataset
from transformers import AutoTokenizer, AutoModel
import torch
import torch.nn.functional as F
```

```
/Library/Frameworks/Python.framework/Versions/3.12/lib/python3.12/site-
```

```
packages/tqdm/auto.py:21: TqdmWarning: IProgress not found. Please update
jupyter and ipywidgets. See
https://ipywidgets.readthedocs.io/en/stable/user_install.html
  from .autonotebook import tqdm as notebook_tqdm
```

1.1.8 Load dataset

[3]:
```
# Load Financial banking77 dataset
train_dataset, test_dataset = load_dataset('banking77', split=['train', 'test'])
print("Train\n", train_dataset)
print("Test\n", test_dataset)

# Accessing example from the training set
example = train_dataset[0]
print("Example from the training set:")
print(example)

# Zhaozhuo: Just organize a bit.
```

```
Train
 Dataset({
    features: ['text', 'label'],
    num_rows: 10003
})
Test
 Dataset({
    features: ['text', 'label'],
    num_rows: 3080
})
Example from the training set:
{'text': 'I am still waiting on my card?', 'label': 11}
```

1.1.9 Load tokenizer

[4]:
```
# Define tokenzier
tokenizer = AutoTokenizer.from_pretrained("sentence-transformers/
 ↪all-MiniLM-L6-v2", use_fast=True)
```

1.1.10 Tokenize dataset

[5]:
```
# Tokenize the training and testing set
tokenized_trainset=tokenizer(train_dataset["text"])['input_ids']
train_labels=train_dataset["label"]

tokenized_testset=tokenizer(test_dataset["text"])['input_ids']
test_labels=test_dataset["label"]
```

1.2 Q1. (40 Points) Write a function to compute Jaccard distance between lists

[6]:
```
def jaccard_distance(list1, list2):
  '''
  Compute the Jaccard distance between two lists.
  '''
  set1 = set(list1)
  set2 = set(list2)
  intersection = len(set1.intersection(set2))
  union = len(set1.union(set2))
  jaccard_similarity = intersection / union
  jaccard_distance = 1 - jaccard_similarity
  return jaccard_distance #return the Jaccard distance
```

1.2.1 Test script to check the Jaccard distance calculation function

[7]:
```
test_distance=jaccard_distance([1,2],[1,3,4])

if(test_distance==0.75):
  print("Q1: Correct")
else:
  print("Q1: Wrong")
```

```
Q1: Correct
```

1.3 Q2(a) (15 Points) Write code to implement the K nearest neighbors (KNN) algorithm with *Jaccard* distance. Use K=1 to compute accuracy on test set.

1.3.1 Note: Please code this section from scratch, do not use KNN implementations from libraries such as scikit-learn.

1.3.2 Hint 1: K=1 means that only one nearest neighbor votes for the sample's label. Find the sample in the trainset which is closest to the test sample and predict it's label as the test sample's label.

1.3.3 Hint 2: Use the tokenized train set and tokenized test set

[8]:
```
total_num_correct = 0
number_of_testing_cases = len(tokenized_testset)

'''
Add KNN code here

'''
def find_nearest_neighbor(train_data, test_instance, k=1):
    distances = []
    for train_instance in train_data:
        dist = jaccard_distance(train_instance[:-1], test_instance[:-1])
```

```
        distances.append((train_instance[-1], dist))
    distances.sort(key=lambda x: x[1])
    nearest_neighbors = [label for label, _ in distances[:k]]
    return max(set(nearest_neighbors), key=nearest_neighbors.count)



for test_instance in tokenized_testset:
    predicted_label = find_nearest_neighbor(tokenized_trainset, test_instance)
    if predicted_label == test_instance[-1]:
        total_num_correct +=1

print("Accuracy:",total_num_correct/number_of_testing_cases)
```

```
Accuracy: 1.0
```

1.4 Q2(b) (15 Points) Please print out three samples from the test set along with their nearest neighbor from the train set. Discuss your observations.

[9]:
```
for i in range(3):
    test_instance = tokenized_testset[i]
    nearest_neighbor = find_nearest_neighbor(tokenized_trainset, test_instance)
    print("Test Instance:", test_instance)
    print("Nearest Neighbor:", nearest_neighbor)
    print()

    '''
from the results we can notice a common label which is 102 is
the most frequent nearest neighbor from the given test instances .
consistent behavior with respect to 'find_nearest_neighbor' function.

'''
```

```
Test Instance: [101, 2129, 2079, 1045, 12453, 2026, 4003, 1029, 102]
Nearest Neighbor: 102

Test Instance: [101, 1045, 2145, 2031, 2025, 2363, 2026, 2047, 4003, 1010, 1045,
3641, 2058, 1037, 2733, 3283, 1012, 102]
Nearest Neighbor: 102

Test Instance: [101, 1045, 3641, 1037, 4003, 2021, 2009, 2038, 2025, 3369, 1012,
2393, 3531, 999, 102]
Nearest Neighbor: 102
```

1.5 Q3(a) (10 Points) Below, we have provided a method to generate embeddings. Please study the code and generate embeddings for the training set. Also complete the function to calculate eucleadian distance

[10]:
```
# Function to calculate Euclidean distance
def Euclidean_distance(array1, array2):
  '''
  Compute the Euclidean distance between two arrays.
  '''
  return torch.sqrt(torch.sum((array1 - array2)**2))# return euclidean distance

#Mean Pooling - Take attention mask into account for correct averaging
def mean_pooling(model_output, attention_mask):
    token_embeddings = model_output[0] #First element of model_output contains␣
 ↪all token embeddings
    input_mask_expanded = attention_mask.unsqueeze(-1).expand(token_embeddings.
 ↪size()).float()
    return torch.sum(token_embeddings * input_mask_expanded, 1) / torch.
 ↪clamp(input_mask_expanded.sum(1), min=1e-9)


# Sentences we want sentence embeddings for
train_sentences = train_dataset["text"]

# Load model from HuggingFace Hub
tokenizer = AutoTokenizer.from_pretrained('sentence-transformers/
 ↪all-MiniLM-L6-v2')
model = AutoModel.from_pretrained('sentence-transformers/all-MiniLM-L6-v2')

# Tokenize sentences
encoded_train_input = tokenizer(train_sentences, padding=True, truncation=True,␣
 ↪return_tensors='pt')

# Compute token embeddings
with torch.no_grad():
    model_output = model(**encoded_train_input)

# Perform pooling
train_embeddings = mean_pooling(model_output,␣
 ↪encoded_train_input['attention_mask'])

# Normalize embeddings
train_embeddings = F.normalize(train_embeddings, p=2, dim=1)

print("train_embeddings:")
print(train_embeddings)
```

```
'''
Write your code here
'''
emb1 = train_embeddings[0]
emb2 = train_embeddings[1]
distance = Euclidean_distance(emb1,emb2)
print("Euclidean distance between embeddings 1 and 2: ", distance.item())
```

```
train_embeddings:
tensor([[-0.0354, -0.0421, -0.0028,  …, -0.1048, -0.0466,  0.0028],
        [ 0.0226, -0.0135,  0.0243,  …,  0.0013, -0.0254,  0.0173],
        [-0.0460, -0.0199, -0.0015,  …, -0.0797,  0.0138,  0.0683],
        …,
        [ 0.0077, -0.0747,  0.0463,  …, -0.0985,  0.0380,  0.0776],
        [ 0.0008,  0.0272, -0.0471,  …, -0.0207,  0.0552,  0.0497],
        [ 0.0956,  0.0028, -0.0104,  …, -0.0103,  0.0139, -0.0248]])
Euclidean distance between embeddings 1 and 2:  0.7619522213935852
```

1.6 Q3(b) (10 Points) Please write a KNN classifier that uses the *Euclidean* distance between *Embeddings* of samples. Predict the labels for the test set and printout the accuracy.

[12]:
```
# Function to calculate Euclidean distance
def Euclidean_distance(array1, array2):
  '''
  Compute the Euclidean distance between two arrays.
  '''
  return torch.sqrt(torch.sum((array1 - array2)**2))# return euclidean distance

#Mean Pooling - Take attention mask into account for correct averaging
def mean_pooling(model_output, attention_mask):
    token_embeddings = model_output[0] #First element of model_output contains␣
 ↪all token embeddings
    input_mask_expanded = attention_mask.unsqueeze(-1).expand(token_embeddings.
 ↪size()).float()
    return torch.sum(token_embeddings * input_mask_expanded, 1) / torch.
 ↪clamp(input_mask_expanded.sum(1), min=1e-9)


# Sentences we want sentence embeddings for
train_sentences = train_dataset["text"]
test_sentences = test_dataset["text"]

# Load model from HuggingFace Hub
tokenizer = AutoTokenizer.from_pretrained('sentence-transformers/
 ↪all-MiniLM-L6-v2')
model = AutoModel.from_pretrained('sentence-transformers/all-MiniLM-L6-v2')
```

```
# Tokenize sentences
encoded_train_input = tokenizer(train_sentences, padding=True, truncation=True,␣
 ↪return_tensors='pt')
encoded_test_input = tokenizer(test_sentences, padding=True, truncation=True,␣
 ↪return_tensors='pt')


# Compute token embeddings
with torch.no_grad():
    model_output = model(**encoded_train_input)

with torch.no_grad():
    model_output_test = model(**encoded_test_input)

# Perform pooling
train_embeddings = mean_pooling(model_output,␣
 ↪encoded_train_input['attention_mask'])
test_embeddings = mean_pooling(model_output_test,␣
 ↪encoded_test_input['attention_mask'])

# Normalize embeddings
train_embeddings = F.normalize(train_embeddings, p=2, dim=1)
test_embeddings = F.normalize(test_embeddings, p=2, dim=1)



total_num_correct = 0
number_of_testing_cases = len(test_embeddings)

'''
Add KNN code here
'''
def find_nearest_neighbor_2(train_data, train_labels, test_instance):
    min_distance = float('inf')
    nearest_neighbor_label = None
    for train_embedding, train_label in zip(train_data, train_labels):
        dist = Euclidean_distance(train_embedding, test_instance)
        if dist < min_distance:
            min_distance = dist
            nearest_neighbor_label = train_label
    return nearest_neighbor_label



for test_instance, true_label in zip(test_embeddings, train_labels):
```

```
    predicted_label = find_nearest_neighbor_2(train_embeddings, train_labels,␣
 ↪test_instance)
    if predicted_label == true_label:
        total_num_correct += 1

print("Accuracy:",total_num_correct/number_of_testing_cases)
```

```
Accuracy: 0.012012987012987014
```

1.7 Q3(c) (10 Points) Printout the same three samples that you used for Q2(b) along with their nearest neighbors found in Q3(b). Discuss your observations.

[19]:
```
for i in range(3):
    test_instance = test_embeddings
    nearest_neighbor = find_nearest_neighbor_2(train_embeddings, train_labels,␣
 ↪test_instance)
    print("Test Instance:", test_instance)
    print("Nearest Neighbor:", nearest_neighbor)
    print()

    '''



'''
```

```
Test Instance: tensor([[-0.0102,  0.0188, -0.0575,  …,  0.0964, -0.0615,
-0.0092],
        [-0.0428, -0.0263,  0.0400,  …, -0.0296, -0.0079,  0.0679],
        [-0.0337,  0.0374,  0.0194,  …,  0.0616, -0.0190,  0.0224],
        …,
        [ 0.1096, -0.0527, -0.0050,  …, -0.1261,  0.0224, -0.0335],
        [ 0.0984, -0.0408,  0.0406,  …, -0.0376,  0.0214, -0.0130],
        [-0.0144,  0.0701, -0.0568,  …, -0.0215,  0.0555,  0.0137]])
Nearest Neighbor: 59

Test Instance: tensor([[-0.0102,  0.0188, -0.0575,  …,  0.0964, -0.0615,
-0.0092],
        [-0.0428, -0.0263,  0.0400,  …, -0.0296, -0.0079,  0.0679],
        [-0.0337,  0.0374,  0.0194,  …,  0.0616, -0.0190,  0.0224],
        …,
        [ 0.1096, -0.0527, -0.0050,  …, -0.1261,  0.0224, -0.0335],
        [ 0.0984, -0.0408,  0.0406,  …, -0.0376,  0.0214, -0.0130],
        [-0.0144,  0.0701, -0.0568,  …, -0.0215,  0.0555,  0.0137]])
Nearest Neighbor: 59

Test Instance: tensor([[-0.0102,  0.0188, -0.0575,  …,  0.0964, -0.0615,
```

```
-0.0092],
        [-0.0428, -0.0263,  0.0400,  …, -0.0296, -0.0079,  0.0679],
        [-0.0337,  0.0374,  0.0194,  …,  0.0616, -0.0190,  0.0224],
        …,
        [ 0.1096, -0.0527, -0.0050,  …, -0.1261,  0.0224, -0.0335],
        [ 0.0984, -0.0408,  0.0406,  …, -0.0376,  0.0214, -0.0130],
        [-0.0144,  0.0701, -0.0568,  …, -0.0215,  0.0555,  0.0137]])
Nearest Neighbor: 59
```