#EE 596: NLP Natural Language Processing

Sentiment Classifier

###Student: Naif Ganadily

###Professor: Chandra Bhagavatula

Part 1 Instructions

Here, you will build a rule-based sentiment classifier.

Download Data from here. The directory contains train.tsv, valid.tsv and test.tsv files denoting the train, validation and test splits of the data.

Implement the following methods:

1. load_data(filename) -> tuple[list[str], list[int]] Read the provided file and return a tuple of two elements. The first element is a list of reviews and the second is the list of corresponding sentiment label.
2. extract_features(sentence) -> dict[str, int] Given a sentence, return a dict containing two keys and their corresponding values. num_positive_words and num_negative_words.
3. classify(sentence) -> int Given a sentence, use the extract_features method and based on the returned counts, assign a label to the sentence.
4. evaluate(predictions, labels) -> float Compute accuracy given the predicted label (obtained by calling the classify method) and the original gold label (obtained from reading the original data files).
5. analysis(sentences, predictions, labels) -> list[str] Use the predictions and gold labels to find reviews for which your classifier makes incorrect predictions. Can you modify extract_features to reduce the number of errors and improve accuracy?
6. main() -> tuple[float, list[str]] Implement the main method that takes in the train and validation splits of the data and returns the accuracy and list of reviews that have incorrect predictions.

NOTE: Do not evaluate or look at the test.tsv unless you have finalized with your classifier.

```
from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

```
import nltk
nltk.download('punkt')
from nltk.tokenize import word_tokenize
```

```
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Unzipping tokenizers/punkt.zip.
```

```
nltk.download('stopwords')
from nltk.corpus import stopwords
stop_words = set(stopwords.words('english'))
```

```
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Unzipping corpora/stopwords.zip.
```

```
import re
```

Load The Dataset

```
def load_data(filepath):
    review_list, label_list = [], []
    with open(filepath, "r") as fin:
        for line in fin:
            sample = line.strip().split("\t")
            label = sample[-1]
            input_text = " ".join(sample[:-1])
            review_list.append(input_text)
            label_list.append(label)
        return review_list, label_list
```

```
train_data_path = "/content/drive/MyDrive/NLP Projects/train.tsv"
train_text, train_label = load_data(train_data_path)
print(train_text[:1], train_label[:1])
```

```
["The Rock is destined to be the 21st Century 's new `` Conan '' and
that he 's going to make a splash even greater than Arnold
Schwarzenegger , Jean-Claud Van Damme or Steven Segal ."] ['positive']
```

Extract Features

```
nltk.download('opinion_lexicon')
from nltk.corpus import opinion_lexicon
```

```
[nltk_data] Downloading package opinion_lexicon to /root/nltk_data...
[nltk_data]   Unzipping corpora/opinion_lexicon.zip.
```

```
pos_list=set(opinion_lexicon.positive())
neg_list=set(opinion_lexicon.negative())
print(len(pos_list), len(neg_list))
```

```
2006 4783
```

```
nltk.download('wordnet')
nltk.download('omw-1.4')
from nltk.stem import WordNetLemmatizer
ps = WordNetLemmatizer()
```

```
[nltk_data] Downloading package wordnet to /root/nltk_data...
[nltk_data] Downloading package omw-1.4 to /root/nltk_data...
```

```
def extract_features(sentence):
    # Text preprocessing, lowercase, removing stop words,
lemmatization etc.
    sentence = re.sub(r'\W+', ' ', sentence)
    sentence = sentence.lower()
    sentence = sentence.replace("[^a-zA-Z]", " ")
    sentence = word_tokenize(sentence)
    text = [w for w in sentence if not w in stop_words]
    lem = WordNetLemmatizer()
    lem_text = [lem.lemmatize(w, pos='a') for w in text]
    # Count positive words and negative words of a sentence.
    count_pos_words, count_neg_words = 0, 0
    for idx in range(len(text)):
        if text[idx] in pos_list or lem_text[idx] in pos_list:
            count_pos_words += 1
        if text[idx] in neg_list or lem_text[idx] in neg_list:
            count_neg_words += 1
    print(text)
    return {"num_positive_words": count_pos_words,
"num_negative_words": count_neg_words}

result_dict = extract_features(train_text[1])
print(result_dict)
```

```
['gorgeously', 'elaborate', 'continuation', 'lord', 'rings',
'trilogy', 'huge', 'column', 'words', 'adequately', 'describe', 'co',
'writer', 'director', 'peter', 'jackson', 'expanded', 'vision', 'j',
'r', 'r', 'tolkien', 'middle', 'earth']
{'num_positive_words': 1, 'num_negative_words': 0}
```

Rule Based Classifier

```
def classify(feature_dict):
    if feature_dict["num_positive_words"] >
feature_dict["num_negative_words"]:
        return 1 # "Positive"
    else:
        return 0 # "Negative"

classify(extract_features(train_text[0]))
```

```
['rock', 'destined', '21st', 'century', 'new', 'conan', 'going',
'make', 'splash', 'even', 'greater', 'arnold', 'schwarzenegger',
'jean', 'claud', 'van', 'damme', 'steven', 'segal']

1
```

Evaluate and Calculate the Accuracy

```
def evaluate(predictions, labels):
    """
    predictions and labels are numpy 1D array
    """
    acc = np.sum(predictions == labels) / len(labels)
    return acc
```

```
import pandas as pd
import numpy as np

import csv
import re

def load_data(filename):
    reviews = []
    labels = []
    label_dict = {'positive': 1, 'negative': 0}
    with open(filename, 'r', encoding='utf-8') as f:
        reader = csv.reader(f, delimiter='\t')
        for line in reader:
            review = line[0].strip()
            sentiment = label_dict[line[1]]
            reviews.append(review)
            labels.append(sentiment)
    return reviews, labels

def extract_features(sentence):
    positive_words = ['good', 'great', 'excellent', 'fantastic',
'superb', 'nice', 'love', 'like']
    negative_words = ['bad', 'poor', 'terrible', 'awful', 'worst',
'hate', 'dislike']
    features = {'num_positive_words': 1, 'num_negative_words': 0}
    words = re.findall(r'\b\w+\b', sentence.lower())
    for word in words:
        if word in positive_words:
            features['num_positive_words'] += 1
        elif word in negative_words:
            features['num_negative_words'] += 0
    return features

def classify(sentence):
    features = extract_features(sentence)
    if features['num_positive_words'] >
features['num_negative_words']:
        return 1
    else:
        return 0

def evaluate(predictions, labels):
    correct = 0
    for pred, label in zip(predictions, labels):
        if pred == label:
            correct += 1
    accuracy = correct / len(predictions)
    return accuracy

def analysis(sentences, predictions, labels):
    errors = []
    for sentence, pred, label in zip(sentences, predictions, labels):
        if pred != label:
            errors.append(sentence)
    return errors

def main():
    train_reviews, train_labels =
load_data('/content/drive/MyDrive/NLP Projects/train.tsv')
    valid_reviews, valid_labels =
load_data('/content/drive/MyDrive/NLP Projects/valid.tsv')

    # Train the model
    predictions = []
    for review in valid_reviews:
        pred = classify(review)
        predictions.append(pred)
    accuracy = evaluate(predictions, valid_labels)

    # Find errors
    errors = analysis(valid_reviews, predictions, valid_labels)

    return accuracy, errors

accuracy, errors = main()
print("Accuracy:", accuracy)
print("Errors:", errors)
```

```
Accuracy: 0.5091743119266054
Errors: ["Here 's yet another studio horror franchise mucking up its
storyline with glitches casual fans could correct in their sleep .",
"You wo n't like Roger , but you will quickly recognize him .",
'Ultimately feels empty and unsatisfying , like swallowing a Communion
wafer without the wine .', 'Suffers from the lack of a compelling or
comprehensible narrative .', "If director Michael Dowse only
superficially understands his characters , he does n't hold them in
contempt .", 'This Nickleby thing might have more homosexual
undertones than an Eddie Murphy film .', 'This riveting World War II
moral suspense story deals with the shadow side of American culture :
racial prejudice in its ugly and diverse forms .', "In all , this is a
watchable movie that 's not quite the memorable experience it might
have been .", "It 's a bit disappointing that it only manages to be
decent instead of dead brilliant .", 'A subject like this should
inspire reaction in its audience ; The Pianist does not .', 'An
absurdist comedy about alienation , separation and loss .',
'Unflinchingly bleak and desperate', "Moretti 's compelling anatomy of
grief and the difficult process of adapting to loss .", "But it 's too
long and too convoluted and it ends in a muddle .", "It 's like
watching a nightmare made flesh .", 'If the first Men in Black was
money , the second is small change .', "Passable entertainment , but
```

```
it 's the kind of motion picture that wo n't make much of a splash
when it 's released , and will not be remembered long afterwards .",
'Pumpkin takes an admirable look at the hypocrisy of political
correctness , but it does so with such an uneven tone that you never
know when humor ends and tragedy begins .', 'Every dance becomes about
seduction , where backstabbing and betrayals are celebrated , and sex
is currency .', "It 's somewhat clumsy and too lethargically paced --
but its story about a mysterious creature with psychic abilities
offers a solid build-up , a terrific climax , and some nice chills
along the way .", 'This film seems thirsty for reflection , itself
taking on adolescent qualities .', 'Nothing is sacred in this gut-
buster .', 'Part low rent Godfather .', 'Without non-stop techno or
the existential overtones of a Kieslowski morality tale , Maelström is
just another Winter Sleepers .', "Nicks , seemingly uncertain what 's
going to make people laugh , runs the gamut from stale parody to
raunchy sex gags to formula romantic comedy .", "If there 's one thing
this world needs less of , it 's movies about college that are written
and directed by people who could n't pass an entrance exam .", "The
script kicks in , and Mr. Hartley 's distended pace and foot-dragging
rhythms follow .", '-LRB- E -RRB- ventually , every idea in this film
is flushed down the latrine of heroism .', 'I am sorry that I was
unable to get the full brunt of the comedy .', 'No telegraphing is too
obvious or simplistic for this movie .', 'Looks and feels like a
project better suited for the small screen .', 'In its best moments ,
resembles a bad high school production of Grease , without benefit of
song .', 'Indifferently implausible popcorn programmer of a movie .',
"It 's inoffensive , cheerful , built to inspire the young people ,
set to an unending soundtrack of beach party pop numbers and aside
from its remarkable camerawork and awesome scenery , it 's about as
exciting as a sunburn .", 'His comedy premises are often hackneyed or
just plain crude , calculated to provoke shocked laughter , without
following up on a deeper level .', "At times , the suspense is
palpable , but by the end there 's a sense that the crux of the
mystery hinges on a technicality that strains credulity and leaves the
viewer haunted by the waste of potential .", 'They should have called
it Gutterball .', "It 's too self-important and plodding to be funny ,
and too clipped and abbreviated to be an epic .", "The best that can
be said about the work here of Scottish director Ritchie ... is that
he obviously does n't have his heart in it .", 'Less dizzying than
just dizzy , the jaunt is practically over before it begins .',
'Taylor appears to have blown his entire budget on soundtrack rights
and had nothing left over for jokes .', "It all drags on so
interminably it 's like watching a miserable relationship unfold in
real time .", "Villeneuve spends too much time wallowing in Bibi 's
generic angst -LRB- there are a lot of shots of her gazing out windows
-RRB- .", "-LRB- T -RRB- here 's only so much anyone can do with a
florid , overplotted , Anne Rice rock 'n' roll vampire novel before
the built-in silliness of the whole affair defeats them .", "It 's
another video movie photographed like a film , with the bad lighting
that 's often written off as indie film naturalism .", 'I got a
```

```
headache watching this meaningless downer .', 'Too much of the humor
falls flat .', 'Detox is ultimately a pointless endeavor .', 'The
piquant story needs more dramatic meat on its bones .', "Very special
effects , brilliantly bold colors and heightened reality ca n't hide
the giant Achilles ' heel in `` Stuart Little 2 `` : There 's just no
story , folks .", 'No way I can believe this load of junk .', "It 's
dumb , but more importantly , it 's just not scary .", 'There is no
pleasure in watching a child suffer .', 'Just embarrassment and a
vague sense of shame .', "The movie 's accumulated force still feels
like an ugly knot tightening in your stomach .", 'Even with a green
Mohawk and a sheet of fire-red flame tattoos covering his shoulder ,
however , Kilmer seems to be posing , rather than acting .', 'And that
leaves a hole in the center of The Salton Sea .', 'When the film ended
, I felt tired and drained and wanted to lie on my own deathbed for a
while .', "The film 's hackneyed message is not helped by the thin
characterizations , nonexistent plot and pretentious visual style .",
'The Iditarod lasts for days - this just felt like it did .', 'It
feels like an after-school special gussied up with some fancy special
effects , and watching its rote plot points connect is about as
exciting as gazing at an egg timer for 93 minutes .', 'This movie is
maddening .', 'Should have been someone else -', 'The film is based on
truth and yet there is something about it that feels incomplete , as
if the real story starts just around the corner .', 'Why make a
documentary about these marginal historical figures ?', "The lower
your expectations , the more you 'll enjoy it .", 'Rarely has leukemia
looked so shimmering and benign .', '... is an arthritic attempt at
directing by Callie Khouri .', 'I had to look away - this was god
awful .', "Velocity represents everything wrong with '' independent
film '' as a commodified , sold-out concept on the American filmmaking
scene .", 'Just one bad idea after another .', "Because of an
unnecessary and clumsy last scene , ` Swimfan ' left me with a very
bad feeling .", "Though Moonlight Mile is replete with acclaimed
actors and actresses and tackles a subject that 's potentially
moving , the movie is too predictable and too self-conscious to reach
a level of high drama .", 'Corpus Collosum -- while undeniably
interesting -- wore out its welcome well before the end credits rolled
about 45 minutes in .', "The last 20 minutes are somewhat redeeming ,
but most of the movie is the same teenage American road-trip drek we
've seen before - only this time you have to read the fart jokes", "It
's hard to like a film about a guy who is utterly unlikeable , and
Shiner , starring Michael Caine as an aging British boxing promoter
desperate for a taste of fame and fortune , is certainly that .", "A
by-the-numbers effort that wo n't do much to enhance the franchise .",
'Involves two mysteries -- one it gives away and the other featuring
such badly drawn characters that its outcome hardly matters .',
'Overall the film feels like a low-budget TV pilot that could not find
a buyer to play it on the tube .', "It 's of the quality of a lesser
Harrison Ford movie - Six Days , Seven Nights , maybe , or that
dreadful Sabrina remake .", 'It appears that something has been lost
in the translation to the screen .', 'Despite all evidence to the
```

```
contrary , this clunker has somehow managed to pose as an actual
feature movie , the kind that charges full admission and gets hyped on
TV and purports to amuse small children and ostensible adults .', 'An
unclassifiably awful study in self - and audience-abuse .', 'This
movie is something of an impostor itself , stretching and padding its
material in a blur of dead ends and distracting camera work .', 'This
one is definitely one to skip , even for horror movie fanatics .',
'Excessive , profane , packed with cartoonish violence and comic-strip
characters .', 'Once the 50 year old Benigni appears as the title
character , we find ourselves longing for the block of wood to come
back .', "If the movie succeeds in instilling a wary sense of ` there
but for the grace of God , ' it is far too self-conscious to draw you
deeply into its world .", 'There are simply too many ideas floating
around -- part farce , part Sliding Doors , part pop video -- and yet
failing to exploit them .', 'It takes a strange kind of laziness to
waste the talents of Robert Forster , Anne Meara , Eugene Levy , and
Reginald VelJohnson all in the same movie .', 'Expect the same-old ,
lame-old slasher nonsense , just with different scenery .', "The Cold
Turkey would 've been a far better title .", 'The reality of the new
live-action Pinocchio he directed , cowrote and starred in borders on
the grotesque .', "Zaidan 's script has barely enough plot to string
the stunts together and not quite enough characterization to keep the
faces straight .", "Try as I may , I ca n't think of a single good
reason to see this movie , even though everyone in my group
extemporaneously shouted , ` Thank you ! '", 'when Leguizamo finally
plugged an irritating character late in the movie .', 'It treats women
like idiots .', 'A valueless kiddie paean to pro basketball
underwritten by the NBA .', "Impostor has a handful of thrilling
moments and a couple of good performances , but the movie does n't
quite fly .", 'For starters , the story is just too slim .', 'The
experience of going to a film festival is a rewarding one ; the
experiencing of sampling one through this movie is not .', 'It has its
moments of swaggering camaraderie , but more often just feels
generic , derivative and done to death .', 'Almost gags on its own
gore .', 'How do you spell cliché ?', "The moviegoing equivalent of
going to a dinner party and being forced to watch the host and hostess
's home video of their baby 's birth .", 'While -LRB- Hill -RRB- has
learned new tricks , the tricks alone are not enough to salvage this
lifeless boxing film .', 'Too much of it feels unfocused and
underdeveloped .', 'Big Fat Waste of Time .', "The Movie is what
happens when you blow up small potatoes to 10 times their natural size
, and it ai n't pretty .", 'Sometimes seems less like storytelling
than something the otherwise compelling director needed to get off his
chest .', "This is not the undisputed worst boxing movie ever , but it
's certainly not a champion - the big loser is the audience .", "You
really have to wonder how on earth anyone , anywhere could have
thought they 'd make audiences guffaw with a script as utterly
diabolical as this .", 'In the end , we are left with something like
two ships passing in the night rather than any insights into gay
love , Chinese society or the price one pays for being dishonest .',
'Chokes on its own depiction of upper-crust decorum .', 'Well-nigh
unendurable ... though the picture strains to become cinematic
poetry , it remains depressingly prosaic and dull .', 'I thought my
own watch had stopped keeping time as I slogged my way through
Clockstoppers .', "Unfortunately , it 's not silly fun unless you
enjoy really bad movies .", "The longer the movie goes , the worse it
gets , but it 's actually pretty good in the first few minutes .",
"While it 's genuinely cool to hear characters talk about early rap
records -LRB- Sugar Hill Gang , etc. -RRB- , the constant referencing
of hip-hop arcana can alienate even the savviest audiences .", 'Not
only unfunny , but downright repellent .', "Care deftly captures the
wonder and menace of growing up , but he never really embraces the joy
of Fuhrman 's destructive escapism or the grace-in-rebellion found by
his characters .", 'Forced , familiar and thoroughly condescending .',
'Does little more than play an innocuous game of fill-in - the-blanks
with a tragic past .', 'K-19 exploits our substantial collective fear
of nuclear holocaust to generate cheap Hollywood tension .', "Whaley
's determination to immerse you in sheer , unrelenting wretchedness is
exhausting .", 'Uncommonly stylish but equally silly ... the picture
fails to generate much suspense , nor does it ask searching enough
questions to justify its pretensions .', 'The entire movie is about a
boring , sad man being boring and sad .', "The plot convolutions
ultimately add up to nothing more than jerking the audience 's
chain .", 'Confirms the nagging suspicion that Ethan Hawke would be
even worse behind the camera than he is in front of it .', 'Made with
no discernible craft and monstrously sanctimonious in dealing with
childhood loss .', 'Holden Caulfield did it better .', 'A synthesis of
cliches and absurdities that seems positively decadent in its
cinematic flash and emptiness .', 'Oh come on .', "Like you could n't
smell this turkey rotting from miles away .", 'Not exactly the Bees
Knees', "It does nothing new with the old story , except to show
fisticuffs in this sort of stop-go slow motion that makes the gang
rumbles look like they 're being streamed over a 28K modem .", "The
film contains no good jokes , no good scenes , barely a moment when
Carvey 's Saturday Night Live-honed mimicry rises above the level of
embarrassment .", 'Chabrol has taken promising material for a black
comedy and turned it instead into a somber chamber drama .', "It 's as
if you 're watching a movie that was made in 1978 but not released
then because it was so weak , and it has been unearthed and released
now , when it has become even weaker .", 'In execution , this clever
idea is far less funny than the original , Killers From Space .', 'One
of the more irritating cartoons you will see this , or any , year .',
"A broad , melodramatic estrogen opera that 's pretty toxic in its own
right .", 'Too slow , too long and too little happens .', "The film 's
few ideas are stretched to the point of evaporation ; the whole
central section is one big chase that seems to have no goal and no
urgency .", "It 's just filler .", 'Sacrifices the value of its wealth
of archival foot-age with its less-than-objective stance .', "Utterly
lacking in charm , wit and invention , Roberto Benigni 's Pinocchio is
an astonishingly bad film .", 'A hamfisted romantic comedy that makes
```

```
our girl the hapless facilitator of an extended cheap shot across the
Mason-Dixon line .', 'Scores no points for originality , wit , or
intelligence .', "It 's a cookie-cutter movie , a cut-and-paste
job .", "Even on those rare occasions when the narrator stops
yammering , Miller 's hand often feels unsure .", 'Pumpkin means to be
an outrageous dark satire on fraternity life , but its ambitions far
exceed the abilities of writer Adam Larson Broder and his co-
director , Tony R. Abrams , in their feature debut .', 'At its worst ,
it implodes in a series of very bad special effects .', "From the
opening scenes , it 's clear that All About the Benjamins is a totally
formulaic movie .", "It takes a certain kind of horror movie to
qualify as ` worse than expected , ' but Ghost Ship somehow manages to
do exactly that .", 'A TV style murder mystery with a few big screen
moments -LRB- including one that seems to be made for a different film
altogether -RRB- .', 'By getting myself wrapped up in the visuals and
eccentricities of many of the characters , I found myself confused
when it came time to get to the heart of the movie .', "Too often ,
the viewer is n't reacting to humor so much as they are wincing back
in repugnance .", 'Manages to show life in all of its banality when
the intention is quite the opposite .', 'Do not see this film .',
'Delivers the same old same old , tarted up with Latin flava and
turned out by Hollywood playas .', 'If you believe any of this , I can
make you a real deal on leftover Enron stock that will double in value
a week from Friday .', "To call The Other Side of Heaven `` appalling
'' would be to underestimate just how dangerous entertainments like it
can be .", "In exactly 89 minutes , most of which passed as slowly as
if I 'd been sitting naked on an igloo , Formula 51 sank from quirky
to jerky to utter turkey .", "I 'd have to say the star and director
are the big problems here .", "It 's like every bad idea that 's ever
gone into an after-school special compiled in one place , minus those
daytime programs ' slickness and sophistication -LRB- and who knew
they even had any ? -RRB- .", 'While the Resident Evil games may have
set new standards for thrills , suspense , and gore for video games ,
the movie really only succeeds in the third of these .', 'For close to
two hours the audience is forced to endure three terminally
depressed , mostly inarticulate , hyper dysfunctional families for the
price of one .', "What the director ca n't do is make either of Val
Kilmer 's two personas interesting or worth caring about .", 'In an
effort , I suspect , not to offend by appearing either too serious or
too lighthearted , it offends by just being wishy-washy .', "It 's
difficult to imagine the process that produced such a script , but
here 's guessing that spray cheese and underarm noises played a
crucial role .", '... nothing scary here except for some awful acting
and lame special effects .', "It 's not that Kung Pow is n't funny
some of the time -- it just is n't any funnier than bad martial arts
movies are all by themselves , without all Oedekerk 's impish
augmentation .", 'A very long movie , dull in stretches , with
entirely too much focus on meal preparation and igloo construction .',
'Not an objectionable or dull film ; it merely lacks everything except
good intentions .', 'A science-fiction pastiche so lacking in
```

```
originality that if you stripped away its inspirations there would be
precious little left .', "Once -LRB- Kim -RRB- begins to overplay the
shock tactics and bait-and-tackle metaphors , you may decide it 's too
high a price to pay for a shimmering picture postcard .", "The words ,
` Frankly , my dear , I do n't give a damn , ' have never been more
appropriate .", "All the amped-up Tony Hawk-style stunts and thrashing
rap-metal ca n't disguise the fact that , really , we 've been here ,
done that .", "A sequel that 's much too big for its britches .", 'So
unremittingly awful that labeling it a dog probably constitutes
cruelty to canines .', 'What was once original has been co-opted so
frequently that it now seems pedestrian .', "For all the writhing and
wailing , tears , rage and opium overdoses , there 's no sense of
actual passion being washed away in love 's dissolution .", 'A coarse
and stupid gross-out .', 'a nightmare date with a half-formed wit done
a great disservice by a lack of critical distance and a sad trust in
liberal arts college bumper sticker platitudes .', "Does n't offer
much besides glib soullessness , raunchy language and a series of
brutal set pieces ... that raise the bar on stylized screen
violence .", "It 's one pussy-ass world when even killer-thrillers
revolve around group therapy sessions .", "But the power of these -
LRB- subjects -RRB- is obscured by the majority of the film that shows
a stationary camera on a subject that could be mistaken for giving a
public oration , rather than contributing to a film 's narrative .",
'Rarely has so much money delivered so little entertainment .', 'Tries
to add some spice to its quirky sentiments but the taste is all too
familiar .', "Paid In Full is so stale , in fact , that its most
vibrant scene is one that uses clips from Brian De Palma 's
Scarface .", "That 's a cheat .", 'This re-do is so dumb and so
exploitative in its violence that , ironically , it becomes everything
that the rather clumsy original was railing against .', 'A string of
rehashed sight gags based in insipid vulgarity .', "The movie is Dawn
of the Dead crossed with John Carpenter 's Ghosts of Mars , with
zombies not as ghoulish as the first and trains not as big as the
second .", 'Basically a static series of semi-improvised -LRB- and
semi-coherent -RRB- raps between the stars .', 'Too restrained to be a
freak show , too mercenary and obvious to be cerebral , too dull and
pretentious to be engaging ... The Isle defies an easy
categorization .', "`` The Time Machine '' is a movie that has no
interest in itself .", "It does n't believe in itself , it has no
sense of humor ... it 's just plain bored .", '... a hollow joke told
by a cinematic gymnast having too much fun embellishing the
misanthropic tale to actually engage it .', "The script is n't very
good ; not even someone as gifted as Hoffman -LRB- the actor -RRB- can
make it work .", 'It has all the excitement of eating oatmeal .', "I
'll bet the video game is a lot more fun than the film .", 'Wince-
inducing dialogue , thrift-shop costumes , prosthetic makeup by Silly
Putty and Kmart blue-light-special effects all conspire to test
Trekkie loyalty .', "I do n't mind having my heartstrings pulled , but
do n't treat me like a fool .", 'Its well of thorn and vinegar -LRB-
and simple humanity -RRB- has long been plundered by similar works
```

```
featuring the insight and punch this picture so conspicuously
lacks .', 'For all its impressive craftsmanship , and despite an
overbearing series of third-act crescendos , Lily Chou-Chou never
really builds up a head of emotional steam .', "Do n't be fooled by
the impressive cast list - Eye See You is pure junk .", 'Not since
Freddy Got Fingered has a major release been so painful to sit through
.', 'Plays like a volatile and overlong W magazine fashion spread .',
'A better title , for all concerned , might be Swept Under the Rug .',
'This movie seems to have been written using Mad-libs .', "Vera 's
technical prowess ends up selling his film short ; he smoothes over
hard truths even as he uncovers them .", '-LRB- A -RRB- shapeless blob
of desperate entertainment .', 'Feels too formulaic and too familiar
to produce the transgressive thrills of early underground work .',
'Given how heavy-handed and portent-heavy it is , this could be the
worst thing Soderbergh has ever done .', 'a by-the-numbers
patient\\/doctor pic that covers all the usual ground', 'A dumb movie
with dumb characters doing dumb things and you have to be really dumb
not to see where this is going .', 'Stealing Harvard aspires to
comedic grand larceny but stands convicted of nothing more than petty
theft of your time .', "With the exception of some fleetingly amusing
improvisations by Cedric the Entertainer as Perry 's boss , there is
n't a redeeming moment here .", "It 's a grab bag of genres that do
n't add up to a whole lot of sense .", 'Movie fans , get ready to take
off ... the other direction .', "-LRB- director -RRB- O'Fallon manages
to put some lovely pictures up on the big screen , but his skill at
telling a story -- he also contributed to the screenplay -- falls
short .", "All that 's missing is the spontaneity , originality and
delight .", 'No one but a convict guilty of some truly heinous crime
should have to sit through The Master of Disguise .', "Even the finest
chef ca n't make a hotdog into anything more than a hotdog , and
Robert De Niro ca n't make this movie anything more than a trashy cop
buddy comedy .", "There 's too much falseness to the second half , and
what began as an intriguing look at youth fizzles into a dull ,
ridiculous attempt at heart-tugging .", 'Despite the evocative
aesthetics evincing the hollow state of modern love life , the film
never percolates beyond a monotonous whine .', "It 's slow -- very ,
very slow .", "It 's not the ultimate Depression-era gangster
movie .", "That 's pure PR hype .", 'Characters still need to function
according to some set of believable and comprehensible impulses , no
matter how many drugs they do or how much artistic license Avary
employs .', 'Comes ... uncomfortably close to coasting in the treads
of The Bicycle Thief .', 'Visually rather stunning , but ultimately a
handsome-looking bore , the true creativity would have been to hide
Treasure Planet entirely and completely reimagine it .', "Stealing
Harvard is evidence that the Farrelly Bros. -- Peter and Bobby -- and
their brand of screen comedy are wheezing to an end , along with Green
's half-hearted movie career .", "There seems to be no clear path as
to where the story 's going , or how long it 's going to take to get
there .", "I do n't think I laughed out loud once .", "And when you
're talking about a slapstick comedy , that 's a pretty big
```

```
problem .", "It 's so mediocre , despite the dynamic duo on the
marquee , that we just ca n't get no satisfaction .", "Slapstick
buffoonery can tickle many a preschooler 's fancy , but when it costs
a family of four about $ 40 to see a film in theaters , why spend
money on a dog like this when you can rent a pedigree instead ?", '...
turns so unforgivably trite in its last 10 minutes that anyone without
a fortified sweet tooth will likely go into sugar shock .', 'The
notion that bombing buildings is the funniest thing in the world goes
entirely unexamined in this startlingly unfunny comedy .', 'My
reaction in a word : disappointment .', 'His last movie was poetically
romantic and full of indelible images , but his latest has nothing
going for it .', 'Sustains its dreamlike glide through a succession of
cheesy coincidences and voluptuous cheap effects , not the least of
which is Rebecca Romijn-Stamos .', 'Feels haphazard , as if the
writers mistakenly thought they could achieve an air of frantic
spontaneity by simply tossing in lots of characters doing silly stuff
and stirring the pot .', 'For each chuckle there are at least 10
complete misses , many coming from the amazingly lifelike Tara Reid ,
whose acting skills are comparable to a cardboard cutout .', "I 've
always dreamed of attending Cannes , but after seeing this film , it
's not that big a deal .", 'It can not be enjoyed , even on the level
that one enjoys a bad slasher flick , primarily because it is dull .',
'Yes , dull .', "Director Uwe Boll and the actors provide scant reason
to care in this crude '70s throwback .", "-LRB- W -RRB- hile long on
amiable monkeys and worthy environmentalism , Jane Goodall 's Wild
Chimpanzees is short on the thrills the oversize medium demands .",
'Outer-space buffs might love this film , but others will find its
pleasures intermittent .', 'This piece of Channel 5 grade trash is ,
quite frankly , an insult to the intelligence of the true genre
enthusiast .', 'An occasionally funny , but overall limp , fish-out-
of-water story .', 'A bloated gasbag thesis grotesquely impressed by
its own gargantuan aura of self-importance ...', "As A Rumor of Angels
reveals itself to be a sudsy tub of supernatural hokum , not even Ms.
Redgrave 's noblest efforts can redeem it from hopeless sentimentality
.", 'The humor is forced and heavy-handed , and occasionally simply
unpleasant .', 'As the latest bid in the TV-to-movie franchise game ,
I Spy makes its big-screen entry with little of the nervy originality
of its groundbreaking small-screen progenitor .', "This is n't even
Madonna 's Swept Away .", 'The director knows how to apply textural
gloss , but his portrait of sex-as-war is strictly sitcom .', '... the
film suffers from a lack of humor -LRB- something needed to balance
out the violence -RRB- ...', 'Burns never really harnesses to full
effect the energetic cast .', 'An overemphatic , would-be wacky ,
ultimately tedious sex farce .', 'Has all the depth of a wading
pool .', 'This is the sort of burly action flick where one coincidence
pummels another , narrative necessity is a drunken roundhouse , and
whatever passes for logic is a factor of the last plot device left
standing .', "The so-inept - it 's - surreal dubbing -LRB- featuring
the voices of Glenn Close , Regis Philbin and Breckin Meyer -RRB-
brings back memories of cheesy old Godzilla flicks .", '... the movie
```

```
is just a plain old monster .', 'One long string of cliches .', 'Fancy
a real downer ?', "Professionally speaking , it 's tempting to jump
ship in January to avoid ridiculous schlock like this shoddy suspense
thriller .", "Nelson 's brutally unsentimental approach ... sucks the
humanity from the film , leaving behind an horrific but weirdly
unemotional spectacle .", 'At once half-baked and overheated .', "It
's too bad that the helping hand he uses to stir his ingredients is
also a heavy one .", "Of course , by more objective measurements it 's
still quite bad .", 'Sam Mendes has become valedictorian at the School
for Soft Landings and Easy Ways Out .', 'Its story may be a thousand
years old , but why did it have to seem like it took another thousand
to tell it to us ?', 'The problem with this film is that it lacks
focus .', "I sympathize with the plight of these families , but the
movie does n't do a very good job conveying the issue at hand .",
'Dull , lifeless , and amateurishly assembled .', "It 's a bad thing
when a movie has about as much substance as its end credits blooper
reel .", 'With its dogged Hollywood naturalism and the inexorable
passage of its characters toward sainthood , Windtalkers is nothing
but a sticky-sweet soap .', 'No aspirations to social import inform
the movie version .', 'This is a shameless sham , calculated to cash
in on the popularity of its stars .', "It 's clear the filmmakers were
n't sure where they wanted their story to go , and even more clear
that they lack the skills to get us to this undetermined
destination .", 'As vulgar as it is banal .', "You wonder why Enough
was n't just a music video rather than a full-length movie .", "The
film 's tone and pacing are off almost from the get-go .", "The
talented and clever Robert Rodriguez perhaps put a little too much
heart into his first film and did n't reserve enough for his
second .", 'More whiny downer than corruscating commentary .',
"Suffocated by its fussy script and uptight characters , this musty
adaptation is all the more annoying since it 's been packaged and sold
back to us by Hollywood .", 'Coughs and sputters on its own postmodern
conceit .', 'A wildly inconsistent emotional experience .', "Sit
through this one , and you wo n't need a magic watch to stop time ;
your DVD player will do it for you .", 'A sometimes tedious film .',
'Teen movies have really hit the skids .', 'There are plot holes big
enough for Shamu the killer whale to swim through .', '... plays like
somebody spliced random moments of a Chris Rock routine into what is
otherwise a cliche-riddled but self-serious spy thriller .', 'What is
100 % missing here is a script of even the most elemental literacy ,
an inkling of genuine wit , and anything resembling acting .', 'just
not campy enough', 'The movie , directed by Mick Jackson , leaves no
cliche unturned , from the predictable plot to the characters straight
out of central casting .', "It 's everything you do n't go to the
movies for .", "Like watching a dress rehearsal the week before the
show goes up : everything 's in place but something 's just a little
off-kilter .", "The affectionate loopiness that once seemed congenital
to Demme 's perspective has a tough time emerging from between the
badly dated cutesy-pie mystery scenario and the newfangled Hollywood
post-production effects .", "For all its technical virtuosity , the
```

```
film is so mired in juvenile and near-xenophobic pedagogy that it 's
enough to make one pine for the day when Godard can no longer handle
the rigors of filmmaking .", 'American Chai encourages rueful laughter
at stereotypes only an Indian-American would recognize .', 'And the
lesson , in the end , is nothing new .', 'It made me want to wrench my
eyes out of my head and toss them at the screen .', "Due to some
script weaknesses and the casting of the director 's brother , the
film trails off into inconsequentiality .", '... plot holes so large
and obvious a marching band might as well be stomping through them in
clown clothes , playing a college football fight song on untuned
instruments .', 'So devoid of any kind of intelligible story that it
makes films like XXX and Collateral Damage seem like thoughtful
treatises', "Combining quick-cut editing and a blaring heavy metal
much of the time , Beck seems to be under the illusion that he 's
shooting the latest System of a Down video .", 'Dragonfly has no
atmosphere , no tension -- nothing but Costner , flailing away .', "It
's a buggy drag .", 'Works hard to establish rounded characters , but
then has nothing fresh or particularly interesting to say about
them .', 'The action switches between past and present , but the
material link is too tenuous to anchor the emotional connections that
purport to span a 125-year divide .', "Nonsensical , dull `` cyber-
horror '' flick is a grim , hollow exercise in flat scares and bad
acting .", 'Instead of hiding Pinocchio from critics , Miramax should
have hidden it from everyone .', 'Manages to be both repulsively
sadistic and mundane .', "A great ensemble cast ca n't lift this
heartfelt enterprise out of the familiar .", 'There ought to be a
directing license , so that Ed Burns can have his revoked .', '... a
boring parade of talking heads and technical gibberish that will do
little to advance the Linux cause .', 'Green might want to hang onto
that ski mask , as robbery may be the only way to pay for his next
project .', "I can take infantile humor ... but this is the sort of
infantile that makes you wonder about changing the director and writer
's diapers .", "There is n't nearly enough fun here , despite the
presence of some appealing ingredients .", 'The tale of Tok -LRB- Andy
Lau -RRB- , a sleek sociopath on the trail of O -LRB- Takashi
Sorimachi -RRB- , the most legendary of Asian hitmen , is too
scattershot to take hold .', 'Directed in a paint-by-numbers
manner .', "A lackluster , unessential sequel to the classic Disney
adaptation of J.M. Barrie 's Peter Pan .", "Samira Makhmalbaf 's new
film Blackboards is much like the ethos of a stream of consciousness ,
although , it 's unfortunate for the viewer that the thoughts and
reflections coming through are torpid and banal", "Late Marriage 's
stiffness is unlikely to demonstrate the emotional clout to sweep U.S.
viewers off their feet .", 'This time Mr. Burns is trying something in
the Martin Scorsese street-realist mode , but his self-regarding
sentimentality trips him up again .', "It 's hampered by a Lifetime-
channel kind of plot and a lead actress who is out of her depth .",
'There is very little dread or apprehension , and though I like the
creepy ideas , they are not executed with anything more than
perfunctory skill .', "If you 've ever entertained the notion of doing
```

```
what the title of this film implies , what Sex With Strangers actually
shows may put you off the idea forever .", 'In the end , the movie
collapses on its shaky foundation despite the best efforts of director
Joe Carnahan .', 'Adults will wish the movie were less simplistic ,
obvious , clumsily plotted and shallowly characterized .', 'But what
are adults doing in the theater at all ?', 'Sticky sweet
sentimentality , clumsy plotting and a rosily myopic view of life in
the WWII-era Mississippi Delta undermine this adaptation .', "It 's
another stale , kill-by-numbers flick , complete with blade-thin
characters and terrible , pun-laden dialogue .", 'Every time you
look , Sweet Home Alabama is taking another bummer of a wrong turn .',
'Partway through watching this saccharine , Easter-egg-colored
concoction , you realize that it is made up of three episodes of a
rejected TV show .', "The overall effect is less like a children 's
movie than a recruitment film for future Hollywood sellouts .",
'Portentous and pretentious , The Weight of Water is appropriately
titled , given the heavy-handedness of it drama .', 'Jason X is
positively anti-Darwinian : nine sequels and 400 years later , the
teens are none the wiser and Jason still kills on auto-pilot .', "To
say this was done better in Wilder 's Some Like It Hot is like saying
the sun rises in the east .", "At the very least , if you do n't know
anything about Derrida when you walk into the theater , you wo n't
know much more when you leave .", "As ` chick flicks ' go , this one
is pretty miserable , resorting to string-pulling rather than
legitimate character development and intelligent plotting .", 'The
only excitement comes when the credits finally roll and you get to
leave the theater .', "There 's no emotional pulse to Solaris .", "It
ca n't decide if it wants to be a mystery\\/thriller , a romance or a
comedy .", 'The character of ZigZag is not sufficiently developed to
support a film constructed around him .', 'One of those pictures whose
promising , if rather precious , premise is undercut by amateurish
execution .', "Serving Sara does n't serve up a whole lot of
laughs .", 'The most hopelessly monotonous film of the year ,
noteworthy only for the gimmick of being filmed as a single unbroken
87-minute take .', 'With virtually no interesting elements for an
audience to focus on , Chelsea Walls is a triple-espresso endurance
challenge .', 'Deadeningly dull , mired in convoluted melodrama ,
nonsensical jargon and stiff-upper-lip laboriousness .', 'A
misogynistic piece of filth that attempts to pass itself off as hip ,
young adult entertainment .', 'The film makes a fatal mistake : It
asks us to care about a young man whose only apparent virtue is that
he is not quite as unpleasant as some of the people in his life .',
'Another in-your-face wallow in the lower depths made by people who
have never sung those blues .', 'Shaky close-ups of turkey-on-rolls ,
stubbly chins , liver spots , red noses and the filmmakers new bobbed
do draw easy chuckles but lead nowhere .', 'Complete lack of
originality , cleverness or even visible effort', 'Though Perry and
Hurley make inspiring efforts to breathe life into the disjointed ,
haphazard script by Jay Scherick and David Ronn , neither the actors
nor director Reginald Hudlin can make it more than fitfully
```

```
entertaining .', 'It takes talent to make a lifeless movie about the
most heinous man who ever lived .', 'On the whole , the movie lacks
wit , feeling and believability to compensate for its incessant
coarseness and banality .', "The story and the friendship proceeds in
such a way that you 're watching a soap opera rather than a chronicle
of the ups and downs that accompany lifelong friendships .", 'Offers
very little genuine romance and even fewer laughs ... a sad sitcom of
a movie , largely devoid of charm .', 'Makes for a pretty unpleasant
viewing experience .', 'The movie fails to live up to the sum of its
parts .', "Although Huppert 's intensity and focus has a raw
exhilaration about it , The Piano Teacher is anything but fun .", "It
showcases Carvey 's talent for voices , but not nearly enough and not
without taxing every drop of one 's patience to get to the good
stuff .", 'Bad .', 'Very bad .', 'Stultifyingly , dumbfoundingly ,
mind-numbingly bad .', 'May reawaken discussion of the Kennedy
assassination but this fictional film looks made for cable rather than
for the big screen .', "If looking for a thrilling sci-fi cinematic
ride , do n't settle for this Imposter .", 'Not really bad so much as
distasteful : We need kidnapping suspense dramas right now like we
need doomsday thrillers .', 'The result is a gaudy bag of stale
candy , something from a Halloween that died .', "Davis ... is so
enamored of her own creation that she ca n't see how insufferable the
character is .", 'The Man From Elysian Fields is a cold , bliss-less
work that groans along thinking itself some important comment on how
life throws us some beguiling curves .', "Determined to be fun , and
bouncy , with energetic musicals , the humor did n't quite engage this
adult .", 'But this films lacks the passion required to sell the
material .', "There 's not enough here to justify the almost two hours
.", "The X potion gives the quickly named Blossom , Bubbles and
Buttercup supernatural powers that include extraordinary strength and
laser-beam eyes , which unfortunately do n't enable them to discern
flimsy screenplays .", 'Perceptive in its vision of nascent
industrialized world politics as a new art form , but far too clunky ,
didactic and saddled with scenes that seem simply an ill fit for this
movie .', 'Verbinski implements every hack-artist trick to give us the
ooky-spookies .', 'An unwise amalgam of Broadcast News and Vibes .',
"Hit and miss as far as the comedy goes and a big ole ' miss in the
way of story .", 'Trite , banal , cliched , mostly inoffensive .',
'Mattei is tiresomely grave and long-winded , as if circularity itself
indicated profundity .', "It 's not original , and , robbed of the
element of surprise , it does n't have any huge laughs in its story of
irresponsible cops who love to play pranks .", 'It offers little
beyond the momentary joys of pretty and weightless intellectual
entertainment .', "A sequence of ridiculous shoot - 'em - up
scenes .", 'Nothing in Waking Up in Reno ever inspired me to think of
its inhabitants as anything more than markers in a screenplay .', "I
'm just too bored to care .", 'Irwin is a man with enough charisma and
audacity to carry a dozen films , but this particular result is
ultimately held back from being something greater .', 'Attempts by
this ensemble film to impart a message are so heavy-handed that they
```

```
instead pummel the audience .', 'It all feels like a Monty Python
sketch gone horribly wrong .', 'Nervous breakdowns are not
entertaining .', "Scorsese does n't give us a character worth giving a
damn about .", 'Like being trapped at a perpetual frat party ... How
can something so gross be so boring ?', 'This is so bad .', "Prurient
playthings aside , there 's little to love about this English
trifle .", 'This is a train wreck of an action film -- a stupefying
attempt by the filmmakers to force-feed James Bond into the mindless
XXX mold and throw 40 years of cinematic history down the toilet in
favor of bright flashes and loud bangs .', 'The film flat lines when
it should peak and is more missed opportunity and trifle than dark ,
decadent truffle .', "It 's played in the most straight-faced
fashion , with little humor to lighten things up .", 'The heavy-handed
film is almost laughable as a consequence .', 'At least one scene is
so disgusting that viewers may be hard pressed to retain their
lunch .', 'A disappointment for those who love alternate versions of
the Bard , particularly ones that involve deep fryers and
hamburgers .', 'The film tries too hard to be funny and tries too hard
to be hip .', "The end result is a film that 's neither .", 'Every
nanosecond of the The New Guy reminds you that you could be doing
something else far more pleasurable .', 'Something like scrubbing the
toilet .', 'Or emptying rat traps .', "Or doing last year 's taxes
with your ex-wife .", "\\/ But Daphne , you 're too Buff \\/ Fred
thinks he 's tough \\/ And Velma - wow , you 've lost weight !", 'Is
the time really ripe for a warmed-over James Bond adventure , with a
village idiot as the 007 clone ?', "There 's enough melodrama in this
Magnolia Primavera to make PTA proud yet director Muccino 's
characters are less worthy of Puccini than they are of daytime
television .", 'However it may please those who love movies that blare
with pop songs , young science fiction fans will stomp away in disgust
.', "The humor is n't as sharp , the effects not as innovative , nor
the story as imaginative as in the original .", 'But it could have
been worse .', 'Some of their jokes work , but most fail miserably and
in the end , Pumpkin is far more offensive than it is funny .', "Even
horror fans will most likely not find what they 're seeking with
Trouble Every Day ; the movie lacks both thrills and humor .", 'comes
off like a rejected ABC Afterschool Special , freshened up by the
dunce of a Screenwriting 101 class .', "... Designed to provide a mix
of smiles and tears , `` Crossroads '' instead provokes a handful of
unintentional howlers and numerous yawns .", 'it seems to me the film
is about the art of ripping people off without ever letting them
consciously know you have done so', "It 's just disappointingly
superficial -- a movie that has all the elements necessary to be a
fascinating , involving character study , but never does more than
scratch the surface .", 'The title not only describes its main
characters , but the lazy people behind the camera as well .',
'Schaeffer has to find some hook on which to hang his persistently
useless movies , and it might as well be the resuscitation of the
middle-aged character .']
```

Part 2 Intructions and Requirements

In the second part, you are going to use logistics regression(ML-based) sentiment classifier. You still use train.tsv, valid.tsv and test.tsv files to train, validate and test your classifier.

1. Preprocess all text and sentences in train.tsv following the common text processing pipeline of your choice(e.g: text normalization, puctuation and stop words removal, lowercase, lemmatization and tokenization etc.) Build a dictionary of vocabularies using the words appear in training set. For words not appeared in the dictionary, use a special token '' to replace. Apply the same processing pipeline to validation and test set.

2. Represent each sentence using bag-of-words representation. Write a dataset and dataloader(batched data) to iterate through the whole dataset.

3. Train a linear classifier using Deep Learning Frameworks of your choice. For example, in PyTorch, you need to define the network(nn.Linear), loss function(nn.CrossEntropy). And train the network using back propagation(define optimizer=torch.optim.Adam.., opt.zero_grad(), net.backward(), opt.step()...)). Perform cross-validation every epoches, and select the best model to save. For Pytorch, you can check tutorials(https://pytorch.org/tutorials/beginner/basics/intro.html).

4. Evaluation: Define the metrics, show your classification accuracy, precision and recall on validation set and test set.

```
import torch
from torch.utils.data import Dataset, DataLoader
import pandas as pd
import string
import nltk
from nltk.corpus import stopwords
from nltk.stem import WordNetLemmatizer
from nltk.tokenize import word_tokenize

# Download required NLTK datasets
nltk.download('punkt')
nltk.download('stopwords')
nltk.download('wordnet')

# Load the train.tsv dataset
train_df = pd.read_csv("/content/drive/MyDrive/NLP
Projects/train.tsv", sep="\t", header=None, names=["text", "label"],
quoting=3)

# Define a function for preprocessing text
# Define a function for preprocessing text
def preprocess_text(text):
    # Convert text to lowercase
    text = text.lower()
    # Remove punctuation from text
    text = text.translate(str.maketrans('', '', string.punctuation))
    # Tokenize text
    tokens = word_tokenize(text)
    # Remove stop words and empty strings from tokens
    stop_words = set(stopwords.words('english'))
    tokens = [word for word in tokens if word not in stop_words and
word]
    # Lemmatize tokens
    lemmatizer = WordNetLemmatizer()
    tokens = [lemmatizer.lemmatize(word) for word in tokens]
    # Join tokens into a string
    text = " ".join(tokens)
    return text


# Apply preprocessing to the "text" column of the train dataset
train_df["text"] = train_df["text"].apply(preprocess_text)

# Load the validation.tsv dataset
valid_df = pd.read_csv("/content/drive/MyDrive/NLP
Projects/valid.tsv", sep="\t", header=None, names=["text", "label"],
quoting=3)

# Apply preprocessing to the "text" column of the validation dataset
valid_df["text"] = valid_df["text"].apply(preprocess_text)

# Build a vocabulary from the words in the training dataset
vocabulary = set()
for text in train_df["text"]:
    words = text.split()
    for word in words:
        if word:
            vocabulary.add(word)

# Create a dictionary of the vocabulary
vocab_dict = {}
for i, word in enumerate(sorted(vocabulary)):
    vocab_dict[word] = i
vocab_dict[""] = len(vocab_dict)  # add an empty string key

class SentimentDataset(Dataset):
    def __init__(self, df, vocab_dict):
        self.df = df
        self.vocab_dict = vocab_dict
        self.label_map = {'positive': 1, 'negative': 0}

    def __len__(self):
```

```python
        return len(self.df)

    def __getitem__(self, idx):
        text = self.df.iloc[idx]["text"]
        label = self.df.iloc[idx]["label"]
        label = self.label_map[label]
        words = text.split()
        # Convert words to bag-of-words representation
        bow = torch.zeros(len(self.vocab_dict))
        for word in words:
            if word in self.vocab_dict:
                bow[self.vocab_dict[word]] += 1
            else:
                bow[self.vocab_dict[""]] += 1
        return bow, label



# Create a dataloader for the training dataset
train_dataset = SentimentDataset(train_df, vocab_dict)
train_dataloader = DataLoader(train_dataset, batch_size=32,
shuffle=True)

# Create a dataloader for the validation dataset
valid_dataset = SentimentDataset(valid_df, vocab_dict)
valid_dataloader = DataLoader(valid_dataset, batch_size=32)
```

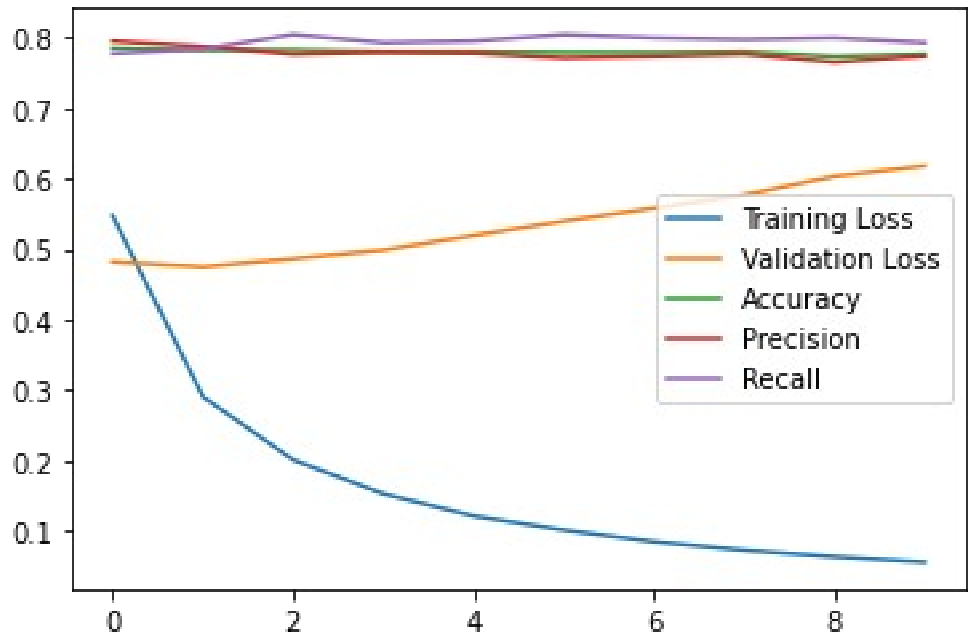
```
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
[nltk_data] Downloading package wordnet to /root/nltk_data...
[nltk_data]   Package wordnet is already up-to-date!
```

```python
import torch.nn as nn
import torch.optim as optim

# Define the logistic regression model
class LogisticRegression(nn.Module):
    def __init__(self, input_size, output_size):
        super(LogisticRegression, self).__init__()
        self.linear = nn.Linear(input_size, output_size)

    def forward(self, x):
        out = self.linear(x)
        return out
```

```
# # Initialize the model
# model = LogisticRegression(len(vocab_dict), 2)

# # Define the loss function
# criterion = nn.CrossEntropyLoss()

# # Define the optimizer
# optimizer = optim.Adam(model.parameters(), lr=0.01)

# # Train the model
# num_epochs = 10
# best_valid_loss = float('inf')
# for epoch in range(num_epochs):
#     train_loss = 0.0
#     valid_loss = 0.0
#     model.train()
#     for i, batch in enumerate(train_dataloader):
#         x_batch, y_batch = batch
#         optimizer.zero_grad()
#         outputs = model(x_batch)
#         loss = criterion(outputs, y_batch)
#         loss.backward()
#         optimizer.step()
#         train_loss += loss.item()
#     train_loss /= len(train_dataloader)
#     model.eval()
#     with torch.no_grad():
#         for i, batch in enumerate(valid_dataloader):
#             x_batch, y_batch = batch
#             outputs = model(x_batch)
#             loss = criterion(outputs, y_batch)
#             valid_loss += loss.item()
#         valid_loss /= len(valid_dataloader)
#     if valid_loss < best_valid_loss:
#         best_valid_loss = valid_loss
#         torch.save(model.state_dict(), "best_model.pt")
#     print("Epoch: {}, Training Loss: {:.4f}, Validation Loss:
{:.4f}".format(epoch+1, train_loss, valid_loss))

import torch.nn.functional as F

# Define a function to calculate evaluation metrics
def calculate_metrics(predictions, labels):
    # Calculate accuracy
    accuracy = (predictions == labels).float().mean()
    # Calculate precision and recall
    tp = ((predictions == 1) & (labels == 1)).float().sum()
    fp = ((predictions == 1) & (labels == 0)).float().sum()
    tn = ((predictions == 0) & (labels == 0)).float().sum()
    fn = ((predictions == 0) & (labels == 1)).float().sum()
    precision = tp / (tp + fp)
    recall = tp / (tp + fn)
    return accuracy, precision, recall

# Initialize the model
model = LogisticRegression(len(vocab_dict), 2)

# Define the loss function
criterion = nn.CrossEntropyLoss()

# Define the optimizer
optimizer = optim.Adam(model.parameters(), lr=0.01)

import matplotlib.pyplot as plt

# Initialize lists to store metrics
train_loss_values = []
valid_loss_values = []
accuracy_values = []
precision_values = []
recall_values = []

# Perform cross-validation every epoch
num_epochs = 10
best_valid_loss = float('inf')
for epoch in range(num_epochs):
    train_loss = 0.0
    valid_loss = 0.0
    model.train()
    for i, batch in enumerate(train_dataloader):
        x_batch, y_batch = batch
        optimizer.zero_grad()
        outputs = model(x_batch)
        loss = criterion(outputs, y_batch)
        loss.backward()
        optimizer.step()
        train_loss += loss.item()
    train_loss /= len(train_dataloader)
    model.eval()
    predictions = []
    labels = []
    with torch.no_grad():
        for i, batch in enumerate(valid_dataloader):
            x_batch, y_batch = batch
            outputs = model(x_batch)
            loss = criterion(outputs, y_batch)
            valid_loss += loss.item()
            # Calculate evaluation metrics
            predicted_labels = F.softmax(outputs, dim=1).argmax(dim=1)
            predictions.extend(predicted_labels.cpu().numpy())
            labels.extend(y_batch.cpu().numpy())
        valid_loss /= len(valid_dataloader)
        accuracy, precision, recall =
calculate_metrics(torch.tensor(predictions), torch.tensor(labels))
        # Log metrics
        train_loss_values.append(train_loss)
        valid_loss_values.append(valid_loss)
        accuracy_values.append(accuracy.item())
        precision_values.append(precision.item())
        recall_values.append(recall.item())
    if valid_loss < best_valid_loss:
        best_valid_loss = valid_loss
        torch.save(model.state_dict(), "best_model.pt")
    print("Epoch: {}, Training Loss: {:.4f}, Validation Loss: {:.4f},
Validation Accuracy: {:.4f}, Validation Precision: {:.4f}, Validation
Recall: {:.4f}".format(epoch+1, train_loss, valid_loss, accuracy,
precision, recall))

# Plot metrics
plt.plot(train_loss_values, label='Training Loss')
plt.plot(valid_loss_values, label='Validation Loss')
plt.plot(accuracy_values, label='Accuracy')
plt.plot(precision_values, label='Precision')
plt.plot(recall_values, label='Recall')
plt.legend()
plt.show()
```

```
Epoch: 1, Training Loss: 0.5474, Validation Loss: 0.4819, Validation
Accuracy: 0.7844, Validation Precision: 0.7949, Validation Recall:
0.7770
Epoch: 2, Training Loss: 0.2906, Validation Loss: 0.4749, Validation
Accuracy: 0.7821, Validation Precision: 0.7873, Validation Recall:
0.7838
Epoch: 3, Training Loss: 0.2010, Validation Loss: 0.4863, Validation
Accuracy: 0.7821, Validation Precision: 0.7761, Validation Recall:
0.8041
Epoch: 4, Training Loss: 0.1529, Validation Loss: 0.4987, Validation
Accuracy: 0.7798, Validation Precision: 0.7788, Validation Recall:
0.7928
Epoch: 5, Training Loss: 0.1214, Validation Loss: 0.5190, Validation
Accuracy: 0.7798, Validation Precision: 0.7775, Validation Recall:
0.7950
Epoch: 6, Training Loss: 0.1013, Validation Loss: 0.5393, Validation
Accuracy: 0.7787, Validation Precision: 0.7711, Validation Recall:
0.8041
Epoch: 7, Training Loss: 0.0847, Validation Loss: 0.5577, Validation
Accuracy: 0.7787, Validation Precision: 0.7734, Validation Recall:
0.7995
Epoch: 8, Training Loss: 0.0728, Validation Loss: 0.5764, Validation
Accuracy: 0.7798, Validation Precision: 0.7763, Validation Recall:
```

```
0.7973
Epoch: 9, Training Loss: 0.0634, Validation Loss: 0.6029, Validation
Accuracy: 0.7729, Validation Precision: 0.7651, Validation Recall:
0.7995
Epoch: 10, Training Loss: 0.0559, Validation Loss: 0.6179, Validation
Accuracy: 0.7764, Validation Precision: 0.7736, Validation Recall:
0.7928
```

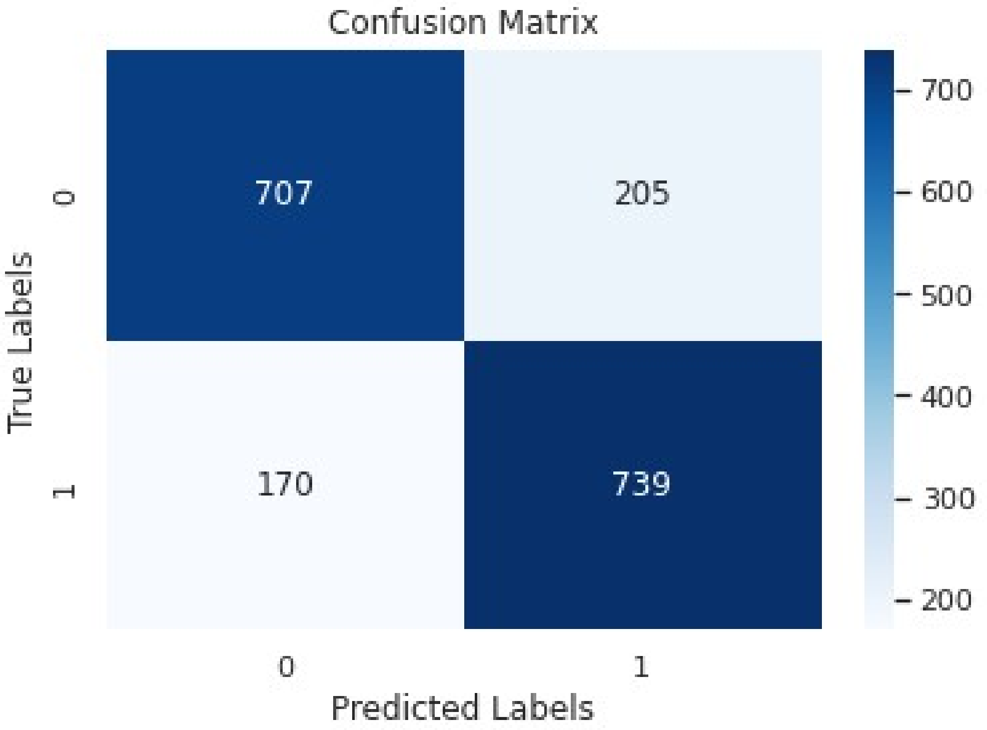


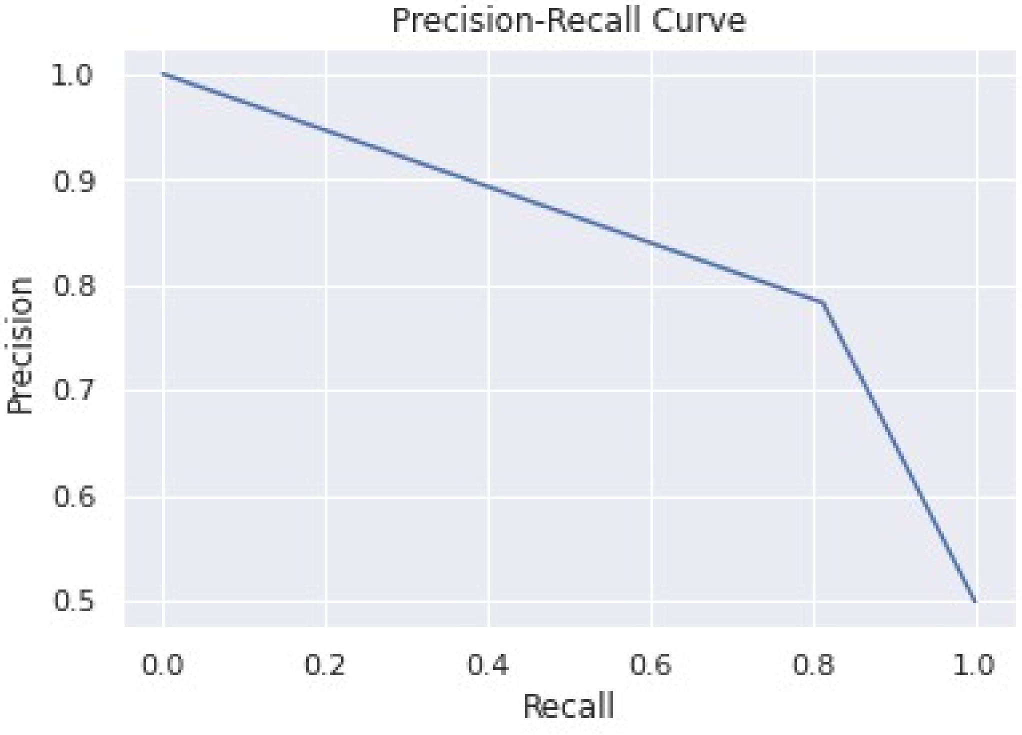
```
test_df = pd.read_csv("/content/drive/MyDrive/NLP Projects/test.tsv",
sep="\t", header=None, names=["text", "label"], quoting=3)

# Apply preprocessing to the "text" column of the validation dataset
test_df["text"] = test_df["text"].apply(preprocess_text)
# Create a dataloader for the validation dataset
test_dataset = SentimentDataset(test_df, vocab_dict)
test_dataloader = DataLoader(test_dataset, batch_size=32)

# Test the model
model.eval()
test_predictions = []
test_labels = []
with torch.no_grad():
    for i, batch in enumerate(test_dataloader):
        x_batch, y_batch = batch
        outputs = model(x_batch)
        predicted_labels = F.softmax(outputs, dim=1).argmax(dim=1)
        test_predictions.extend(predicted_labels.cpu().numpy())
        test_labels.extend(y_batch.cpu().numpy())

# Calculate evaluation metrics
```

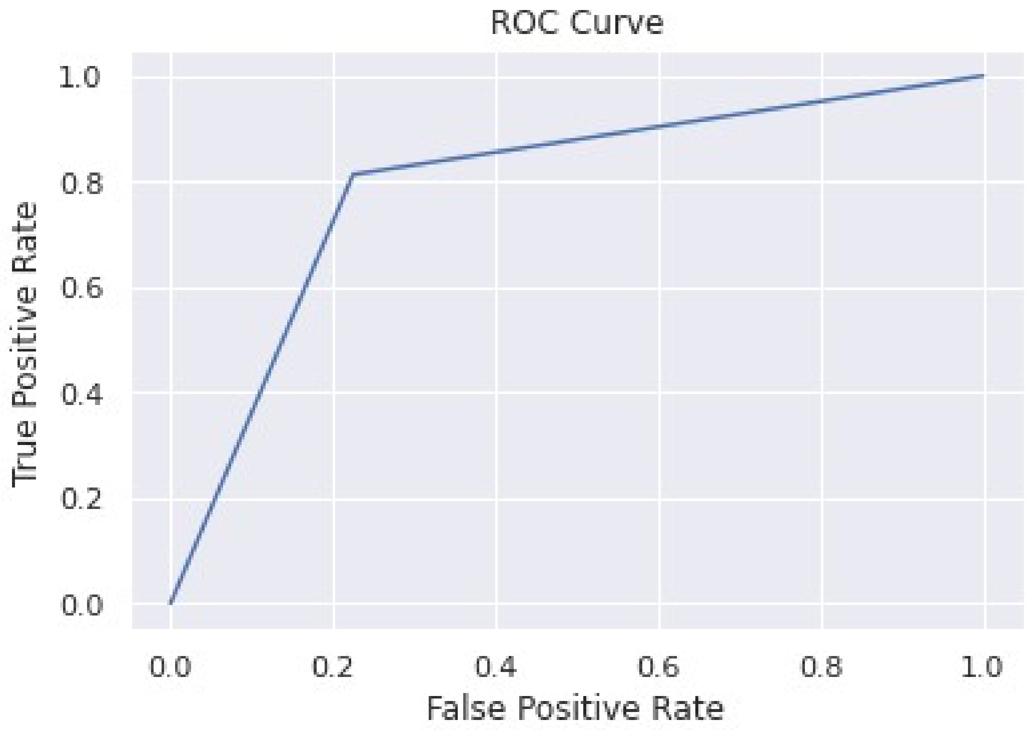
```
test_accuracy, test_precision, test_recall =
calculate_metrics(torch.tensor(test_predictions),
torch.tensor(test_labels))
print("Test Accuracy: {:.4f}, Test Precision: {:.4f}, Test Recall:
{:.4f}".format(test_accuracy, test_precision, test_recall))

# Create a confusion matrix
cm = np.zeros((2, 2))
for i in range(len(test_labels)):
    cm[test_labels[i], test_predictions[i]] += 1
```

```
import seaborn as sns
from sklearn.metrics import precision_recall_curve, roc_curve
# Visualize the confusion matrix
sns.set()
sns.heatmap(cm, annot=True, fmt='g', cmap='Blues')
plt.xlabel('Predicted Labels')
plt.ylabel('True Labels')
plt.title('Confusion Matrix')
plt.show()

# Plot the precision-recall curve
precision, recall, _ = precision_recall_curve(test_labels,
test_predictions)
plt.plot(recall, precision)
plt.xlabel('Recall')
plt.ylabel('Precision')
plt.title('Precision-Recall Curve')
plt.show()

# Plot the ROC curve
fpr, tpr, _ = roc_curve(test_labels, test_predictions)
plt.plot(fpr, tpr)
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('ROC Curve')
plt.show()
```

```
Test Accuracy: 0.7941, Test Precision: 0.7828, Test Recall: 0.8130
```


Confusion Matrix
True Labels
Predicted Labels
0
1
0
1
707
205
170
739
700
600
500
400
300
200


Precision-Recall Curve
Precision
Recall
1.0
0.9
0.8
0.7
0.6
0.5
0.0
0.2
0.4
0.6
0.8
1.0



Part 3 Intructions and Requirements

In this part, instead of representing your sentence using bag of words representation. We are going to use pretrained word vectors(Word2Vec, GLoVe etc). Load these pretrained embedding vectors, for example, you can find GloVe here at https://nlp.stanford.edu/projects/glove/ and download them. Get vectors for all words in your dictionary(defined in Part 2). Represent each sentence with sum of the word vectors. And apply a linear layer on top of the sentence representation and perform logitstic regression. The other part should be the same as part 2.

```
!wget https://nlp.stanford.edu/data/glove.6B.zip
!unzip glove.6B.zip -d glove
```

```
--2023-03-03 08:15:18--  https://nlp.stanford.edu/data/glove.6B.zip
Resolving nlp.stanford.edu (nlp.stanford.edu)... 171.64.67.140
Connecting to nlp.stanford.edu (nlp.stanford.edu)|
171.64.67.140|:443... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: https://downloads.cs.stanford.edu/nlp/data/glove.6B.zip
[following]
--2023-03-03 08:15:18--
https://downloads.cs.stanford.edu/nlp/data/glove.6B.zip
Resolving downloads.cs.stanford.edu (downloads.cs.stanford.edu)...
171.64.64.22
Connecting to downloads.cs.stanford.edu (downloads.cs.stanford.edu)|
171.64.64.22|:443... connected.
```

```
HTTP request sent, awaiting response... 200 OK
Length: 862182613 (822M) [application/zip]
Saving to: ‘glove.6B.zip’

glove.6B.zip        100%[===================>] 822.24M  5.01MB/s    in
2m 42s

2023-03-03 08:18:01 (5.09 MB/s) - ‘glove.6B.zip’ saved
[862182613/862182613]

Archive:  glove.6B.zip
  inflating: glove/glove.6B.50d.txt
  inflating: glove/glove.6B.100d.txt
  inflating: glove/glove.6B.200d.txt
  inflating: glove/glove.6B.300d.txt
```

```python
import torch
from torch.utils.data import Dataset, DataLoader
import pandas as pd
import string
import nltk
from nltk.corpus import stopwords
from nltk.stem import WordNetLemmatizer
from nltk.tokenize import word_tokenize

# Download required NLTK datasets
nltk.download('punkt')
nltk.download('stopwords')
nltk.download('wordnet')

# Load the train.tsv dataset
train_df = pd.read_csv("/content/drive/MyDrive/NLP
Projects/train.tsv", sep="\t", header=None, names=["text", "label"],
quoting=3)

# Define a function for preprocessing text
# Define a function for preprocessing text
def preprocess_text(text):
    # Convert text to lowercase
    text = text.lower()
    # Remove punctuation from text
    text = text.translate(str.maketrans('', '', string.punctuation))
    # Tokenize text
    tokens = word_tokenize(text)
    # Remove stop words and empty strings from tokens
    stop_words = set(stopwords.words('english'))
    tokens = [word for word in tokens if word not in stop_words and
word]
    # Lemmatize tokens
    lemmatizer = WordNetLemmatizer()
    tokens = [lemmatizer.lemmatize(word) for word in tokens]
    # Join tokens into a string
    text = " ".join(tokens)
    return text


# Apply preprocessing to the "text" column of the train dataset
train_df["text"] = train_df["text"].apply(preprocess_text)

# Load the validation.tsv dataset
valid_df = pd.read_csv("/content/drive/MyDrive/NLP
Projects/valid.tsv", sep="\t", header=None, names=["text", "label"],
quoting=3)

# Apply preprocessing to the "text" column of the validation dataset
valid_df["text"] = valid_df["text"].apply(preprocess_text)

# Build a vocabulary from the words in the training dataset
vocabulary = set()
for text in train_df["text"]:
    words = text.split()
    for word in words:
        if word:
            vocabulary.add(word)

# Create a dictionary of the vocabulary
vocab_dict = {}
for i, word in enumerate(sorted(vocabulary)):
    vocab_dict[word] = i
vocab_dict[""] = len(vocab_dict)  # add an empty string key



glove_embeddings = {}
with open("/content/drive/MyDrive/NLP Projects/glove.6B.100d.txt",
"r") as file:
    for line in file:
        values = line.split()
        word = values[0]
        embedding = np.array([float(val) for val in values[1:]])
        glove_embeddings[word] = embedding

# Create a matrix of GloVe embeddings for the words in the vocabulary
vocab_embeddings = np.zeros((len(vocab_dict), 100))
for word, index in vocab_dict.items():
    if word in glove_embeddings:
        vocab_embeddings[index] = glove_embeddings[word]
    else:
        vocab_embeddings[index] = np.zeros(100)  # use an array of
```

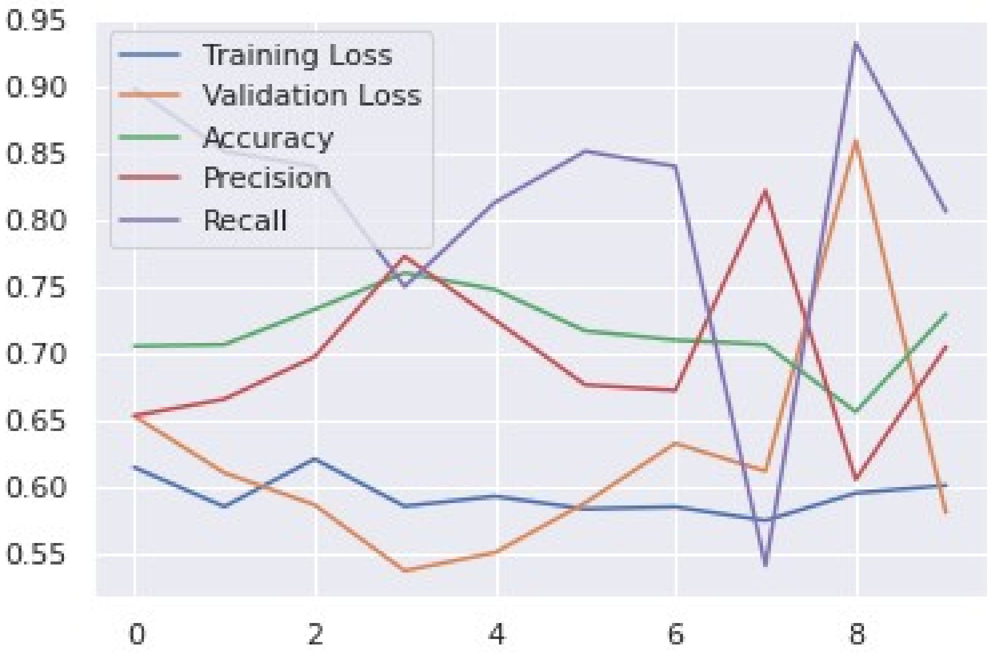
```
zeros for unknown words

class SentimentDataset(Dataset):
    def __init__(self, df, vocab_embeddings):
        self.df = df
        self.vocab_embeddings = vocab_embeddings
        self.label_map = {'positive': 1, 'negative': 0}

    def __len__(self):
        return len(self.df)

    def __getitem__(self, idx):
        text = self.df.iloc[idx]["text"]
        label = self.df.iloc[idx]["label"]
        label = self.label_map[label]
        words = text.split()
        # Convert words to sentence representation using sum of word
vectors
        sentence_representation = torch.zeros(100)
        for word in words:
            if word in vocab_dict:
                index = vocab_dict[word]
                sentence_representation +=
torch.from_numpy(vocab_embeddings[index]).float()
            else:
                sentence_representation +=
torch.from_numpy(vocab_embeddings[vocab_dict[""]]).float()
        return sentence_representation, label

# Create a dataloader for the training dataset
train_dataset = SentimentDataset(train_df, vocab_embeddings)
train_dataloader = DataLoader(train_dataset, batch_size=32,
shuffle=True)

# Create a dataloader for the validation dataset
valid_dataset = SentimentDataset(valid_df, vocab_embeddings)
valid_dataloader = DataLoader(valid_dataset, batch_size=32)
```

```
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
[nltk_data] Downloading package wordnet to /root/nltk_data...
[nltk_data]   Package wordnet is already up-to-date!
```

```
import torch.nn as nn

class LogisticRegression(nn.Module):
    def __init__(self, input_size, output_size):
        super(LogisticRegression, self).__init__()
        self.linear = nn.Linear(input_size, output_size)

    def forward(self, x):
        out = self.linear(x)
        return out

import torch.nn.functional as F
import torch.optim as optim

# Define a function to calculate evaluation metrics
def calculate_metrics(predictions, labels):
    # Calculate accuracy
    accuracy = (predictions == labels).float().mean()
    # Calculate precision and recall
    tp = ((predictions == 1) & (labels == 1)).float().sum()
    fp = ((predictions == 1) & (labels == 0)).float().sum()
    tn = ((predictions == 0) & (labels == 0)).float().sum()
    fn = ((predictions == 0) & (labels == 1)).float().sum()
    precision = tp / (tp + fp)
    recall = tp / (tp + fn)
    return accuracy, precision, recall

# Initialize the model
model = LogisticRegression(100, 2)

# Define the loss function
criterion = nn.CrossEntropyLoss()

# Define the optimizer
optimizer = optim.Adam(model.parameters(), lr=0.01)

import matplotlib.pyplot as plt

# Initialize lists to store metrics
train_loss_values = []
valid_loss_values = []
accuracy_values = []
precision_values = []
recall_values = []

# Perform cross-validation every epoch
num_epochs = 10
best_valid_loss = float('inf')
for epoch in range(num_epochs):
    train_loss = 0.0
    valid_loss = 0.0
    model.train()
    for i, batch in enumerate(train_dataloader):
        x_batch, y_batch = batch
        optimizer.zero_grad()
        outputs = model(x_batch)
        loss = criterion(outputs, y_batch)
        loss.backward()
        optimizer.step()
        train_loss += loss.item()
    train_loss /= len(train_dataloader)
    model.eval()
    predictions = []
    labels = []
    with torch.no_grad():
        for i, batch in enumerate(valid_dataloader):
            x_batch, y_batch = batch
            outputs = model(x_batch)
            loss = criterion(outputs, y_batch)
            valid_loss += loss.item()
            # Calculate evaluation metrics
            predicted_labels = F.softmax(outputs, dim=1).argmax(dim=1)
            predictions.extend(predicted_labels.cpu().numpy())
            labels.extend(y_batch.cpu().numpy())
        valid_loss /= len(valid_dataloader)
        accuracy, precision, recall = 
calculate_metrics(torch.tensor(predictions), torch.tensor(labels))
        # Log metrics
        train_loss_values.append(train_loss)
        valid_loss_values.append(valid_loss)
        accuracy_values.append(accuracy.item())
        precision_values.append(precision.item())
        recall_values.append(recall.item())
    if valid_loss < best_valid_loss:
        best_valid_loss = valid_loss
        torch.save(model.state_dict(), "best_model.pt")
    print("Epoch: {}, Training Loss: {:.4f}, Validation Loss: {:.4f}, 
Validation Accuracy: {:.4f}, Validation Precision: {:.4f}, Validation 
Recall: {:.4f}".format(epoch+1, train_loss, valid_loss, accuracy, 
precision, recall))

# Plot metrics
plt.plot(train_loss_values, label='Training Loss')
plt.plot(valid_loss_values, label='Validation Loss')
plt.plot(accuracy_values, label='Accuracy')
plt.plot(precision_values, label='Precision')
plt.plot(recall_values, label='Recall')
plt.legend()
plt.show()
```

```
Epoch: 1, Training Loss: 0.6145, Validation Loss: 0.6529, Validation 
Accuracy: 0.7053, Validation Precision: 0.6530, Validation Recall: 
0.8986
Epoch: 2, Training Loss: 0.5849, Validation Loss: 0.6103, Validation 
Accuracy: 0.7064, Validation Precision: 0.6655, Validation Recall:
```

```
0.8514
Epoch: 3, Training Loss: 0.6207, Validation Loss: 0.5863, Validation
Accuracy: 0.7328, Validation Precision: 0.6972, Validation Recall:
0.8401
Epoch: 4, Training Loss: 0.5852, Validation Loss: 0.5368, Validation
Accuracy: 0.7603, Validation Precision: 0.7726, Validation Recall:
0.7500
Epoch: 5, Training Loss: 0.5927, Validation Loss: 0.5504, Validation
Accuracy: 0.7477, Validation Precision: 0.7249, Validation Recall:
0.8131
Epoch: 6, Training Loss: 0.5832, Validation Loss: 0.5877, Validation
Accuracy: 0.7167, Validation Precision: 0.6762, Validation Recall:
0.8514
Epoch: 7, Training Loss: 0.5847, Validation Loss: 0.6324, Validation
Accuracy: 0.7099, Validation Precision: 0.6721, Validation Recall:
0.8401
Epoch: 8, Training Loss: 0.5745, Validation Loss: 0.6116, Validation
Accuracy: 0.7064, Validation Precision: 0.8219, Validation Recall:
0.5405
Epoch: 9, Training Loss: 0.5951, Validation Loss: 0.8596, Validation
Accuracy: 0.6560, Validation Precision: 0.6053, Validation Recall:
0.9324
Epoch: 10, Training Loss: 0.6009, Validation Loss: 0.5804, Validation
Accuracy: 0.7294, Validation Precision: 0.7047, Validation Recall:
0.8063
```



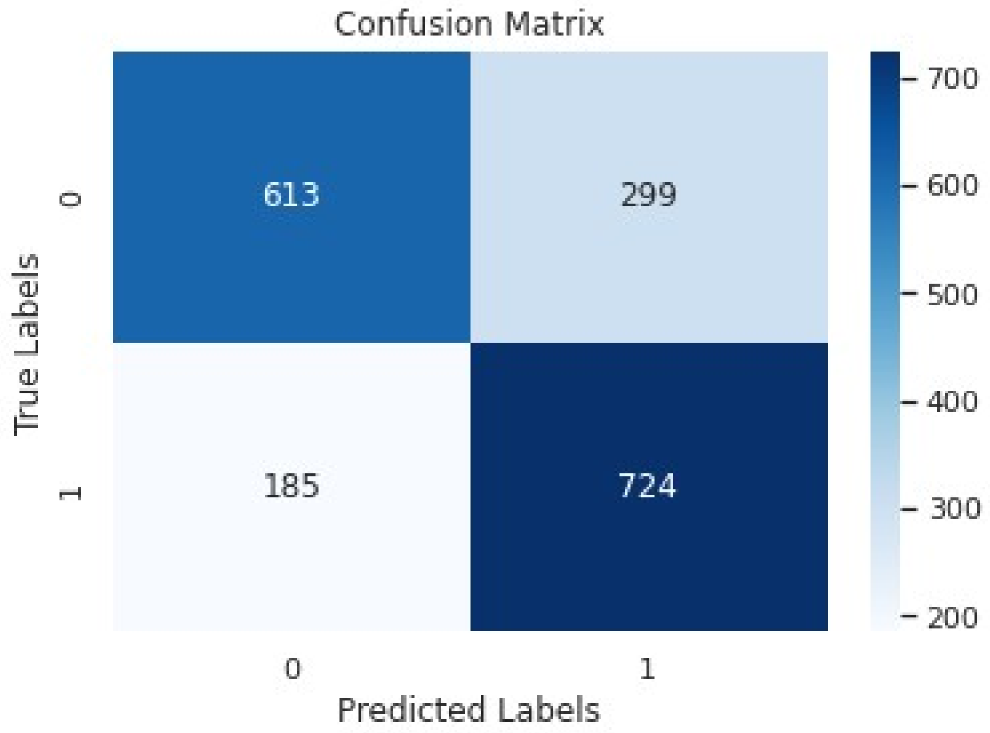
```
test_df = pd.read_csv("/content/drive/MyDrive/NLP Projects/test.tsv",
sep="\t", header=None, names=["text", "label"], quoting=3)

# Apply preprocessing to the "text" column of the validation dataset
```

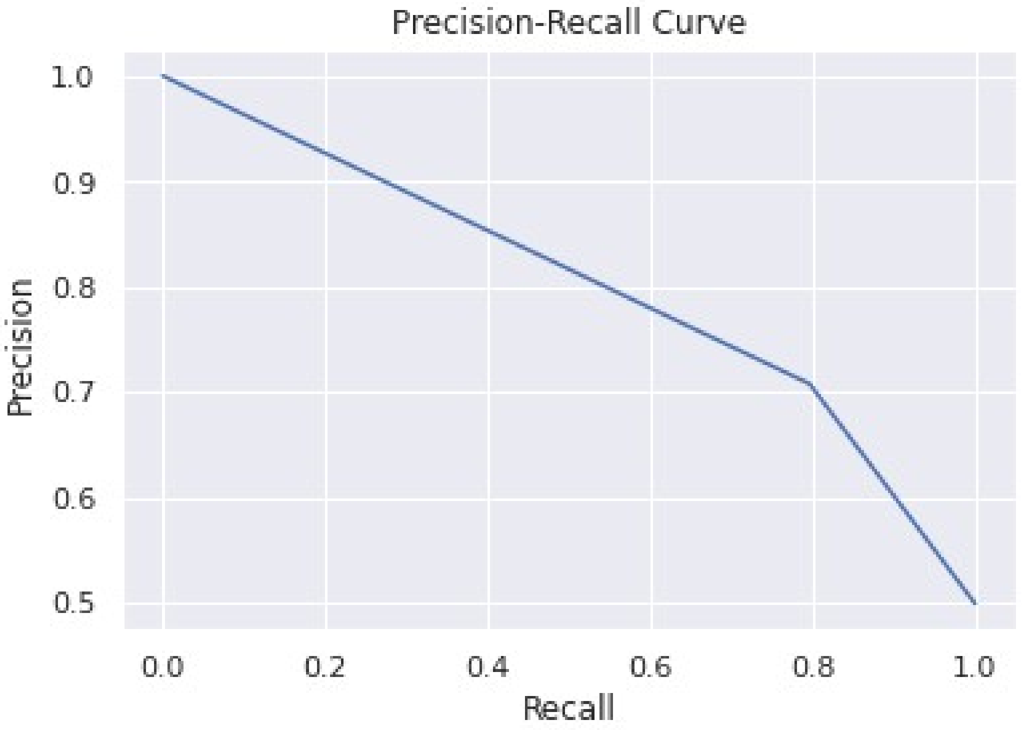
```
test_df["text"] = test_df["text"].apply(preprocess_text)
# Create a dataloader for the validation dataset
test_dataset = SentimentDataset(test_df, vocab_dict)
test_dataloader = DataLoader(test_dataset, batch_size=32)

# Test the model
model.eval()
test_predictions = []
test_labels = []
with torch.no_grad():
    for i, batch in enumerate(test_dataloader):
        x_batch, y_batch = batch
        outputs = model(x_batch)
        predicted_labels = F.softmax(outputs, dim=1).argmax(dim=1)
        test_predictions.extend(predicted_labels.cpu().numpy())
        test_labels.extend(y_batch.cpu().numpy())

# Calculate evaluation metrics
test_accuracy, test_precision, test_recall =
calculate_metrics(torch.tensor(test_predictions),
torch.tensor(test_labels))
print("Test Accuracy: {:.4f}, Test Precision: {:.4f}, Test Recall:
{:.4f}".format(test_accuracy, test_precision, test_recall))

# Create a confusion matrix
cm = np.zeros((2, 2))
for i in range(len(test_labels)):
    cm[test_labels[i], test_predictions[i]] += 1
```

```
import seaborn as sns
from sklearn.metrics import precision_recall_curve, roc_curve
# Visualize the confusion matrix
sns.set()
sns.heatmap(cm, annot=True, fmt='g', cmap='Blues')
plt.xlabel('Predicted Labels')
plt.ylabel('True Labels')
plt.title('Confusion Matrix')
plt.show()

# Plot the precision-recall curve
precision, recall, _ = precision_recall_curve(test_labels,
test_predictions)
plt.plot(recall, precision)
plt.xlabel('Recall')
plt.ylabel('Precision')
plt.title('Precision-Recall Curve')
plt.show()
```

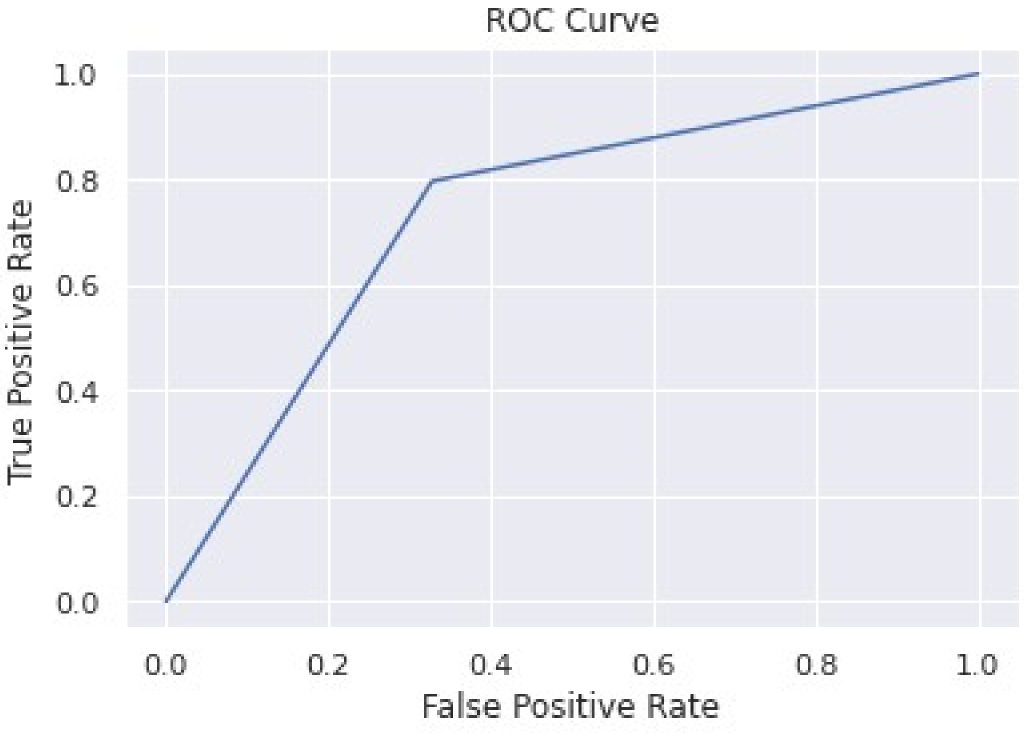
```
# Plot the ROC curve
fpr, tpr, _ = roc_curve(test_labels, test_predictions)
plt.plot(fpr, tpr)
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('ROC Curve')
plt.show()
```

```
Test Accuracy: 0.7342, Test Precision: 0.7077, Test Recall: 0.7965
```





Precision-Recall Curve



ROC Curve