Jason Park
ECE 473
Homework 4

Problem 1

1.

a. From the hinge loss definition: $Loss_{hinge}(x, y, \boldsymbol{w}) = \max\{0, 1 - \boldsymbol{w} * \phi(x)y\}$

We can calculate the gradient piecewise: $\nabla_{\boldsymbol{w}} Loss_{hinge} = \{\, 0, if\ \boldsymbol{w} * Cy \geq 1$
$1, else\ \}$

We also want to run sgd, updating the weights according to: $\boldsymbol{w_{new}} \leftarrow w_{old} - \eta \nabla_{\boldsymbol{w}} Loss_{hinge}$

The problem also required that we use $\eta = .5$

(-1) pretty bad

$\phi(x) = \{pretty: 1, good: 0, bad: 1, plot: 0, not: 0, scenery: 0\}$

y	$\boldsymbol{w}_{old}$	$\nabla_{\boldsymbol{w}} Loss_{hinge}$	$\boldsymbol{W}_{new}$
-1	[0, 0, 0, 0, 0, 0]	[1, 0, 1, 0, 0, 0]	[-.5, 0, -.5, 0, 0, 0]

(+1) good plot

$\phi(x) = \{pretty: 0, good: 1, bad: 0, plot: 1, not: 0, scenery: 0\}$

y	$\boldsymbol{w}_{old}$	$\nabla_{\boldsymbol{w}} Loss_{hinge}$	$\boldsymbol{W}_{new}$
+1	[-.5, 0, -.5, 0, 0, 0]	[0, 1, 0, 1, 0, 0]	[-.5, .5, -.5, .5, 0, 0]

(-1) not good

$\phi(x) = \{pretty: 0, good: 1, bad: 0, plot: 0, not: 1, scenery: 0\}$

y	$\boldsymbol{w}_{old}$	$\nabla_{\boldsymbol{w}} Loss_{hinge}$	$\boldsymbol{W}_{new}$
-1	[-.5, .5, -.5, .5, 0, 0]	[0, 1, 0, 0, 1, 0]	[-.5, 0, -.5, .5, -.5, 0]

(+1) pretty scenery

$\phi(x) = \{pretty: 1, good: 0, bad: 0, plot: 0, not: 0, scenery: 1\}$

y	$\boldsymbol{w}_{old}$	$\nabla_{\boldsymbol{w}} Loss_{hinge}$	$\boldsymbol{W}_{new}$
1	[-.5, 0, -.5, .5, -.5, 0]	[1, 0, 0, 0, 0, 1]	[0, 0, -.5, .5, -.5, .5]

Final weight vector: [0, 0, -.5, .5, -.5, .5]

b. A = {not: w_1, bad: w_2, good: w_3}
Dataset = {good: $w_3 > 0$, not good: $w_1 + w_3 < 0$, bad: $w_2 < 0$, not bad: $w_1 + w_2 > 0$}

Based on this data set, no feature can get zero error because we have a contradiction, based on $w_1 + w_2 > 0$ and $w_1 + w_3 < 0$. Because we have already defined $w_3 > 0$ and $w_2 < 0$, we run into issues with w_1 somehow being > 0 and < 0. Adding the additional feature not bad: w_4 to A would mean that we could remove and avoid this contradiction. The new dataset would be {good: $w_3 > 0$, not good: $w_1 + w_3 < 0$, bad: $w_2 < 0$, not bad: $w_4 > 1$}, for example.

Jason Park
ECE 473
Homework 4

<u>Problem 2</u>

2. d
   1. For the prediction below, I believe the classifier would need more information for bite and humor, because most of the other words are less common or unique to each individual review.

```
=== screenwriter dan schneider and director shawn levy substitute volume and primary colors for humor and bite .
Truth: -1, Prediction: 1 [WRONG]
and                           3 * 0.4600000000000001 = 1.3800000000000003
humor                         1 * 0.9900000000000007 = 0.9900000000000007
director                      1 * 0.21000000000000002 = 0.21000000000000002
screenwriter                  1 * 0.01 = 0.01
substitute                    1 * 0.01 = 0.01
dan                           1 * 0 = 0
shawn                         1 * 0 = 0
volume                        1 * 0 = 0
primary                       1 * 0 = 0
.                             1 * -0.05999999999999114 = -0.05999999999999114
levy                          1 * -0.19000000000000003 = -0.19000000000000003
colors                        1 * -0.21000000000000005 = -0.21000000000000005
schneider                     1 * -0.3900000000000002 = -0.3900000000000002
bite                          1 * -0.3900000000000002 = -0.3900000000000002
for                           1 * -0.4900000000000019 = -0.4900000000000019
```

   2. For the prediction below, I believe the classifier would need more information on words like ever, epic, and be, whose weights are very high and would need more data to make a more accurate prediction.

```
=== a searing , epic treatment of a nationwide blight that seems to be , horrifyingly , ever on the rise .
Truth: 1, Prediction: -1 [WRONG]
ever                          1 * 0.9900000000000007 = 0.9900000000000007
epic                          1 * 0.9800000000000006 = 0.9800000000000006
rise                          1 * 0.5900000000000003 = 0.5900000000000003
a                             2 * 0.20000000000000456 = 0.4000000000000091
of                            1 * 0.3600000000000043 = 0.3600000000000043
,                             3 * 0.11999999999999993 = 0.35999999999999976
treatment                     1 * 0.21000000000000005 = 0.21000000000000005
nationwide                    1 * 0 = 0
blight                        1 * 0 = 0
horrifyingly                  1 * 0 = 0
searing                       1 * -0.01 = -0.01
.                             1 * -0.05999999999999114 = -0.05999999999999114
the                           1 * -0.24000000000001512 = -0.24000000000001512
to                            1 * -0.5500000000000024 = -0.5500000000000024
that                          1 * -0.6300000000000003 = -0.6300000000000003
on                            1 * -0.8700000000000006 = -0.8700000000000006
be                            1 * -0.9100000000000006 = -0.9100000000000006
seems                         1 * -1.0000000000000007 = -1.0000000000000007
```

Jason Park
ECE 473
Homework 4

3. This classifier prediction below would more information on words like film and lacks because the weights are very high and need more data to be properly weighed.

```
=== a perfectly competent and often imaginative film that lacks what little lilo & stitch had in spades -- charisma .
Truth: 1, Prediction: -1 [WRONG]
film                        1 * 0.9800000000000006 = 0.9800000000000006
what                        1 * 0.6100000000000002 = 0.6100000000000002
perfectly                   1 * 0.5900000000000003 = 0.5900000000000003
&                           1 * 0.5900000000000003 = 0.5900000000000003
and                         1 * 0.4600000000000001 = 0.4600000000000001
imaginative                 1 * 0.4100000000000002 = 0.4100000000000002
charisma                    1 * 0.4100000000000002 = 0.4100000000000002
a                           1 * 0.20000000000000456 = 0.20000000000000456
lilo                        1 * 0.19000000000000003 = 0.19000000000000003
stitch                      1 * 0.19000000000000003 = 0.19000000000000003
often                       1 * 0.18999999999999984 = 0.18999999999999984
spades                      1 * 0 = 0
.                           1 * -0.05999999999999114 = -0.05999999999999114
competent                   1 * -0.3900000000000002 = -0.3900000000000002
--                          1 * -0.42000000000000004 = -0.42000000000000004
in                          1 * -0.5000000000000023 = -0.5000000000000023
that                        1 * -0.6300000000000003 = -0.6300000000000003
had                         1 * -0.9500000000000006 = -0.9500000000000006
little                      1 * -0.9700000000000006 = -0.9700000000000006
lacks                       1 * -1.0000000000000007 = -1.0000000000000007
```

4. This classifier would need more information on ride, modern, through, which, and to, which are all words that are weighed very high and may be throwing off the prediction.

```
=== a heady , biting , be-bop ride through nighttime manhattan , a loquacious videologue of the modern male and the lengths to which he'll go to weave a protective cocoon around his own
Truth: 1, Prediction: -1 [WRONG]
ride                        1 * 1.0000000000000007 = 1.0000000000000007
modern                      1 * 0.9900000000000007 = 0.9900000000000007
a                           3 * 0.20000000000000456 = 0.6000000000000136
and                         1 * 0.4600000000000001 = 0.4600000000000001
of                          1 * 0.3600000000000043 = 0.3600000000000043
,                           3 * 0.11999999999999993 = 0.35999999999999976
own                         1 * 0.21000000000000021 = 0.21000000000000021
he'll                       1 * 0.19000000000000003 = 0.19000000000000003
ego                         1 * 0.19000000000000003 = 0.19000000000000003
heady                       1 * 0.01 = 0.01
be-bop                      1 * 0 = 0
nighttime                   1 * 0 = 0
loquacious                  1 * 0 = 0
videologue                  1 * 0 = 0
lengths                     1 * 0 = 0
weave                       1 * 0 = 0
protective                  1 * 0 = 0
cocoon                      1 * 0 = 0
.                           1 * -0.05999999999999114 = -0.05999999999999114
his                         1 * -0.18999999999999942 = -0.18999999999999942
biting                      1 * -0.19000000000000003 = -0.19000000000000003
around                      1 * -0.19000000000000022 = -0.19000000000000022
manhattan                   1 * -0.21000000000000005 = -0.21000000000000005
the                         2 * -0.24000000000001512 = -0.48000000000003024
male                        1 * -0.5900000000000003 = -0.5900000000000003
go                          1 * -0.8100000000000005 = -0.8100000000000005
which                       1 * -0.9300000000000006 = -0.9300000000000006
through                     1 * -0.9700000000000006 = -0.9700000000000006
to                          2 * -0.5500000000000024 = -1.1000000000000048
```

Jason Park
ECE 473
Homework 4

5. This classifier would benefit from having more information on words like films and sweet, as they are weighed very high and may be throwing off the prediction.

```
=== 'it's painful to watch witherspoon's talents wasting away inside unnecessary films like legally blonde and sweet home abomination , i mean , alabama . '
Truth: -1, Prediction: 1 [WRONG]
films                          1 * 1.0000000000000007 = 1.0000000000000007
sweet                          1 * 0.9800000000000006 = 0.9800000000000006
painful                        1 * 0.5900000000000003 = 0.5900000000000003
inside                         1 * 0.5900000000000003 = 0.5900000000000003
and                            1 * 0.4600000000000001 = 0.4600000000000001
home                           1 * 0.3900000000000002 = 0.3900000000000002
,                              2 * 0.11999999999999993 = 0.23999999999999985
watch                          1 * 0.19000000000000014 = 0.19000000000000014
legally                        1 * 0.19000000000000003 = 0.19000000000000003
blonde                         1 * 0.19000000000000003 = 0.19000000000000003
alabama                        1 * 0.19000000000000003 = 0.19000000000000003
away                           1 * 0.010000000000000004 = 0.010000000000000004
'it's                          1 * 0 = 0
witherspoon's                  1 * 0 = 0
wasting                        1 * 0 = 0
abomination                    1 * 0 = 0
.                              1 * -0.05999999999999114 = -0.05999999999999114
talents                        1 * -0.3900000000000002 = -0.3900000000000002
mean                           1 * -0.3900000000000002 = -0.3900000000000002
to                             1 * -0.5500000000000024 = -0.5500000000000024
unnecessary                    1 * -0.5900000000000003 = -0.5900000000000003
like                           1 * -0.9100000000000006 = -0.9100000000000006
i                              1 * -0.9300000000000006 = -0.9300000000000006
'                              1 * -0.9300000000000006 = -0.9300000000000006
```

Jason Park
ECE 473
Homework 4

3. 2f
   I would explain the similar error to the way both feature extractors work. Using stochastic gradient descent is very similar (in learning) to a simpler method, which involves just mapping each string of n characters to the number of times it occurs. If I were to try to construct a review, I would try to say something along the lines of, “The movie did a pretty good job of managing actors and a pretty good job with the plot,” using the logic that n-gram classification would handle duplicates better than word features.

Jason Park
ECE 473
Homework 4

Problem 3

a. Adding the 'entity is' feature string before every feature seems to have drastically minimized the error we received from the data. The new feature tries to find features in the data to see if they are a valid entity, and if we have predicted them correctly. The validation example that is correct classified correct entities like the senate and Margaret McCullough, which without entity is were incorrect. The learned features helped from the evidence that the training and validation error rate were significantly lower, and also upon visual inspection of the error analysis, we can tell that it is doing better at correctly recognizing entities.
b. The new features added were the "left is" and the "right is". An example being instead of just Eduardo Romero, we now read , Eduardo Romero (. I believe it was now able to correctly classify Amman and Andre Caboche, but it also now incorrectly predicted ". Margaret McCullough ,". The learning features helped because the validation error rate was now lowered from the before the two features were added.
c. The newest feature added this time around was checking through the entities inside the entity produced from 3a. For example, instead of saying "entity contains: Eduardo Romero" we now can check "entity contains: Eduardo". Validation examples like Felix Mantilla are now correct because of the new feature, along with the Kurdistan Workers Party. They are both now correct apparently primarily because of the new features. Besides having new correct validation examples, the training error rate was still low, and then the test error continued to improve with the addition of the features in 3c.
d. The new feature breaks the word entity in half, into a prefix and suffix. A good example for this would be in "entity is Sarah Pitkowski". We now also have the entity containing: "Sarah, suffix arah, Pitkowski, suffix wski, and prefix Pitk" This adds more variation to the data, and in turn, more features to learn from. While I couldn't find any specific examples, I can say that there were definitely features that were now correctly predicted, because doing a basic "ctrl-f" to find the number of times WRONG occurs, we can see that part d had less occurrences of incorrect values than part c. Additionally, we can say that the learned features helped because we saw an improvement in validation test error again.