CS57800 Statistical Machine Learning
Homework 2

Penghao Wang

wang1128@purdue.edu

October 7, 2015

1 Foundations

1.1

Linear function: $x_1 + x_2 + x_3 + x_4 >= 2$

Boolean function : $(x_1 \wedge x_2) \vee (x_1 \wedge x_3) \vee (x_1 \wedge x_4) \vee (x_2 \wedge x_3) \vee (x_2 \wedge x_4) \vee (x_3 \wedge x_4)$

1.2

The size of the CON_B is : 2^n

1.3

$\beta_n = \beta_0 + y_i u_i$
If: $\beta_n = \beta^*$
Then: $0 \leq 0$
If: $\beta_n \neq \beta^*$
$\beta^* = \beta_n + \sum y_i u_i$
$y_i \beta' u_i < 0$ Because the β_n update, we know
$y_i \beta' u_i < 0$
$y_i (\beta_0 + y_i u_i)' u_i < 0$
$y_i \beta_0 u_i + 1 < 0$
$y_i \beta_0 u_i < -1$
Because the β is the final vector, we know:
$y_i \beta^* u_i > 0$

$y_i(\beta_n + \sum y_i u_i)u_i > 0$

$y_i\beta_n u_i + y_i(\sum y_i u_i)u_i > 0$

because we have know the $y_i\beta_0 u_i < -1$

we know : $y_i(\sum y_i u_i)u_i > 1$

we want to know : $\|\beta_n - \beta^*\| \leq \|\beta_0 - \beta^*\| - 1$

which is :$y_i(\beta_0 - \beta^*)u_i \leq -1$

which is: $y_i\beta_0 u_i - y_i\beta^* u_i \leq -1$

we know : $y_i\beta^{*'}u_i \geq 1$ and $y_i\beta_0 u_i - y_i < 0$

then it is proved

It is proved.

1.4

Find-s algorithm is mistake bound algorithm for the for learning Boolean conjunctions.

1. First initialize hypothesis h in H.

2. H is $(x_1 \wedge \neg x_1) \wedge (x_2 \wedge \neg x_2) \wedge (x_3 \wedge \neg x_3) \wedge (x_4 \wedge \neg x_5) \wedge (x_5 \wedge \neg x_5) \wedge (x_6 \wedge \neg x_6) \wedge (x_7 \wedge \neg x_7) \wedge ...$

3. For each attribute z in h: if the constraint in h satisfied by x then do nothing, if not remove the the attribute z in the h and generate new constraint that is satisfied by x.

4. For example, if the value of x_1 is 1, then the h $\neg x_1$ will be removed.

5. The h may be $(x_1) \wedge (x_2 \wedge \neg x_2) \wedge (x_3 \wedge \neg x_3) \wedge (x_4 \wedge \neg x_5) \wedge (x_5 \wedge \neg x_5) \wedge (x_6 \wedge \neg x_6) \wedge (x_7 \wedge \neg x_7) \wedge ...$

5. Then repeat the step 3.

6. If we are unlucky, we need $n - 1$ step to find the h in the H space. That means, at most the algorithm will make N-1 mistakes.

7. In that case, This algorithm is a mistake bound algorithm for Boolean conjunctions.

1.5

Both of the classifier will converge. The training error will be 0 and 0. Because the classiffiers are converge, there are no training error.

1.6

When $y' \neq y$, the perceptron update.

The original one could be converted as:

$\|\sum_{i\epsilon N} y_i x_i\|^2 \leq \sum_{i\epsilon N} \|x_i\|^2$

because the y_i stands for the sign of the x_i, if perceptron update, it means the y_i are not in the same value, which is When $y' \neq y$, the perceptron update. In that case, the perceptron update.

So, when perceptron update, if it only update for one time(the first time), $\|\sum_{i\epsilon N} y_i x_i\|^2 = \sum_{i\epsilon N} \|x_i\|^2$ holds for all time.

When perceptron update,$w_i = w_i + y_i x_i$

then: $\|w_i\|^2 = \left\|w_(i-1)\right\|^2 + 2w_{i-1Z}(x_iy_i) + \|x_i\|^2$
we know: $2w_{i-1}(x_iy_i) < 0$ because it updates.
Then $\|w_i\|^2 \leq \|w_{i-1}\|^2 + \|x_i\|^2$
Then $\|w_N\|^2 \leq \|w_0\|^2 + \|\sum_{i\epsilon N} x_i\|^2$
When it updates by a sequence of training instances, we know
$\|\sum_{i\epsilon N} y_ix_i\|^2 = \|w_N\|^2 \leq \|\sum_{i\epsilon N} x_i\|^2$

2 Programming Report

2.1 Introduction

In the homework, two online learning algorithms are implemented, the Perceptron algorithm and the Winnow algorithm. The observation of their performances in practice by running the algorithms over the movie review datasets.

2.2 Feature sets

In the perceptron and winnow algorithm. X is a vector that contains 0 and 1. In the beginning, the list of words are created. Then, an array that contains zeros are created. Then If the word in the first row of the review is in the word list, then find the index of the position of the word and change value of that position from 0 to 1. For example, the first word in the first review is "message". The length of the list of the word is 5000. And the message appears in the first position, the the array will be (1,0,0,0,0...,0)

2.3 Fuctions in the project

1. The numpy and csv function are imported. Numpy is used to calculate the perceptron and winnow algorithm.

2. $cal\_reviewlistlabel()$ function is used to read the csv file. The parameter is filename. It return $review_list$, $review_label$. For the label part, if the label is '+', the review label is 1. Otherwise, the review label will be -1.

3. $calListuniWord(review\_list)$ function is used to generate the list of uni-gram words and save them to the $list\_word$. When read the unigram word to the list, the frequency of the words are calculated. I delete the words that appears less then five time or more than 300 times. I do this because the most frequent words are the words like : the, a , film, an... These words are meaningless. Also, the words that appear in low frequency are deleted because they won't affect the weight too much.

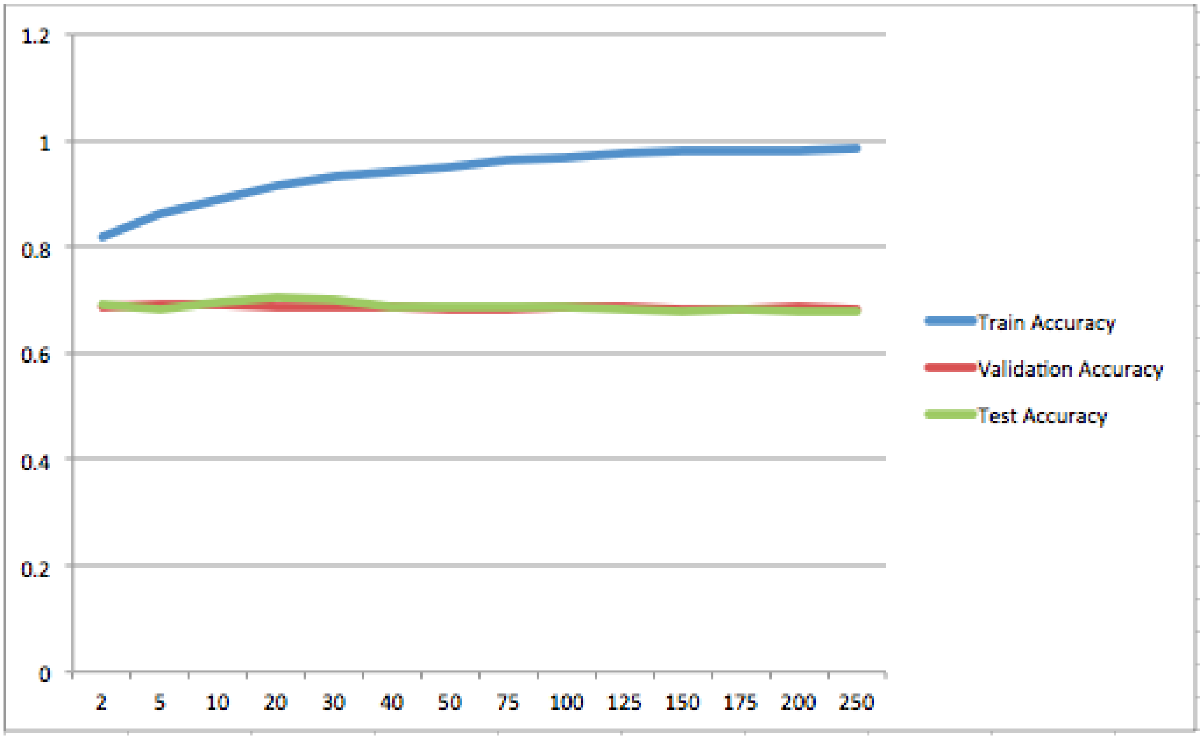
4. $calListbiWord(review\_list)$: function is used to generate the list of bi-gram words. It is pretty like $calListuniWord(review\_list)$. It is also remove the most common words and the words that appear only few times.

5. $cal\_feature\_array(review\_list, list\_word)$ $cal\_bifeature\_array$ and $cal\_both\_feature\_array$: is calculate the unigram,bigram and both feature set.
6. perceptron() function is calculate w by using perceptron algorithm and feature sets.
7. winnow() function is calculate w by using winnow algorithm and feature sets.
8. $calPrecsion\_p()$ is to calculate precision.
9. $calRecall\_p()$ is to calculate recall.
10. $calFscore()$ is to calculate F-score
11. $calTrainError\_p()$ is to calculate accuracy
12. calaprf() is to print out the precision, recall , Fscore and accuracy

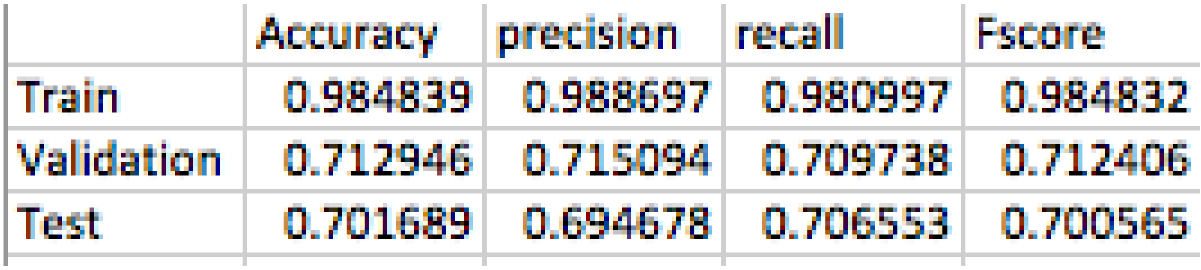
2.4 Performance

2.4.1 Perceptron

Unigram: The following picture shows the accuracy by using Perceptron algorithm. The word list is unigram words. We could know that, the accuracy of the train data is relative high when compare with the validation accuracy. When the maxIter goes up, the accuracy of Perceptron algorithm increase. However, validation accuracy is a little bit decrease. It could be overfit training.

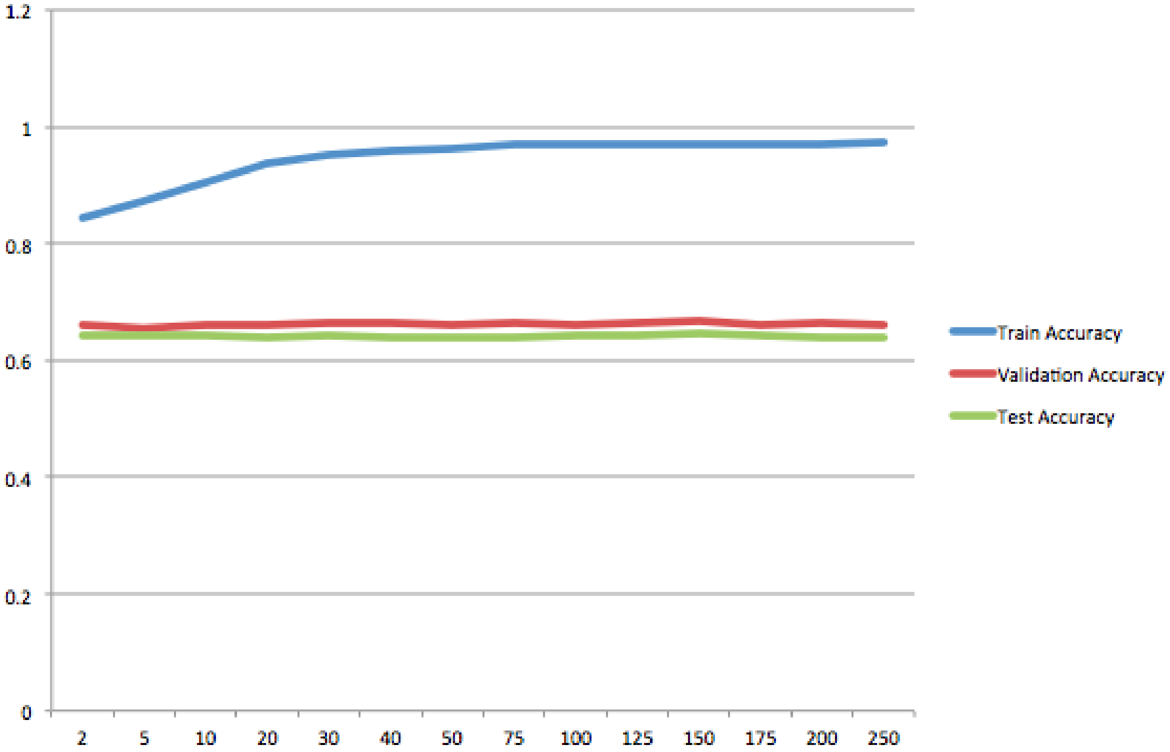


From, the validation accuracy, we find that the best maxIter is 10. The precision,recall and F-score of train data, validation data and test data are in following picture.

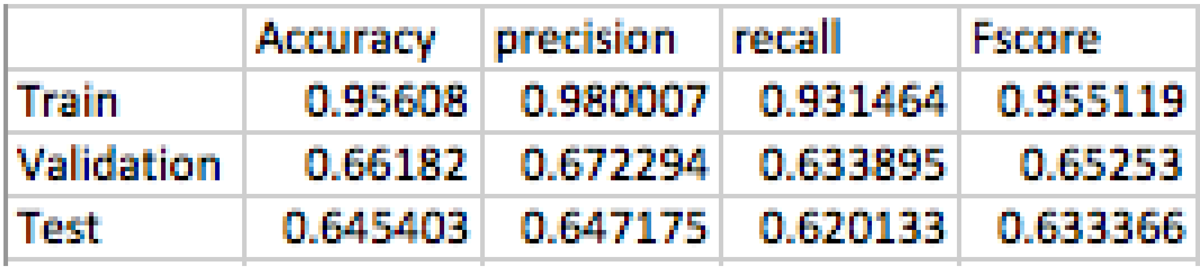


	Accuracy	precision	recall	Fscore
Train	0.984839	0.988697	0.980997	0.984832
Validation	0.712946	0.715094	0.709738	0.712406
Test	0.701689	0.694678	0.706553	0.700565

Bigram: The following picture shows the accuracy by using Perceptron algorithm. The word list is bigram words. We could know that, the accuracy of the train data is relative high when compare with the validation accuracy. When the maxIter goes up, the accuracy of Perceptron algorithm increase. However, validation accuracy is a little bit decrease. It could be overfit training.



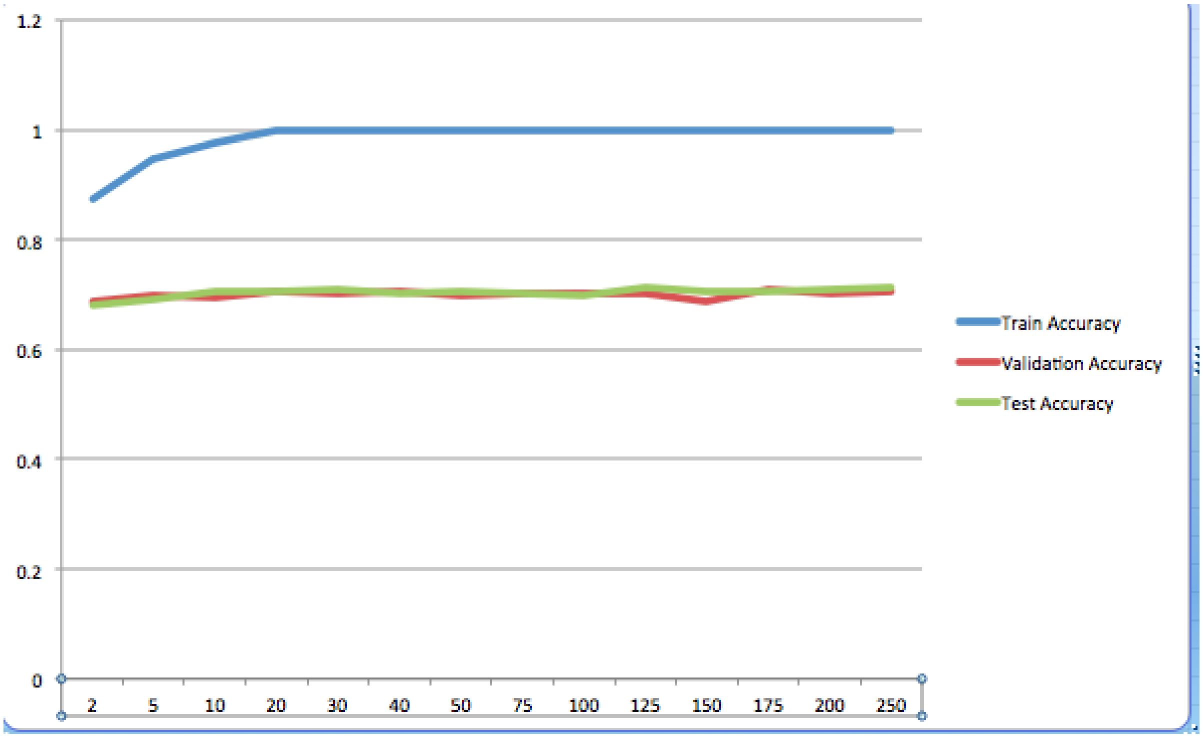
From, the validation accuracy, we find that the best maxIter is 10. The precision,recall and F-score of train data, validation data and test data are in following picture. Something interesting is, even it converge, the accuracy of validation is not the highest.



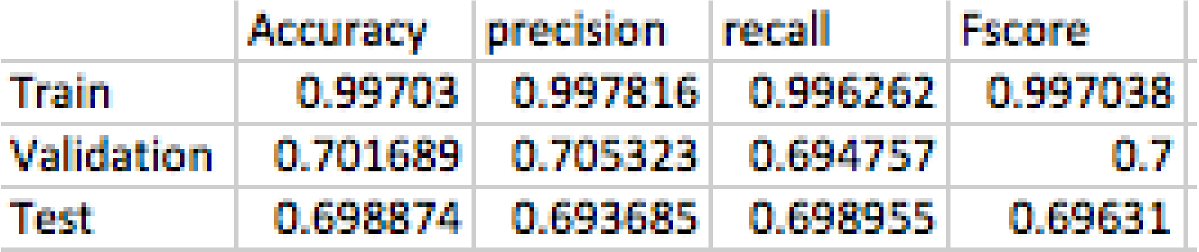
	Accuracy	precision	recall	Fscore
Train	0.95608	0.980007	0.931464	0.955119
Validation	0.66182	0.672294	0.633895	0.65253
Test	0.645403	0.647175	0.620133	0.633366

Both of unigram and bigram: The following picture shows the accuracy by using Perceptron algorithm. It almost same as unigram and a little better than bigram. It converge in about 30.

However, it affect the validation data a little.



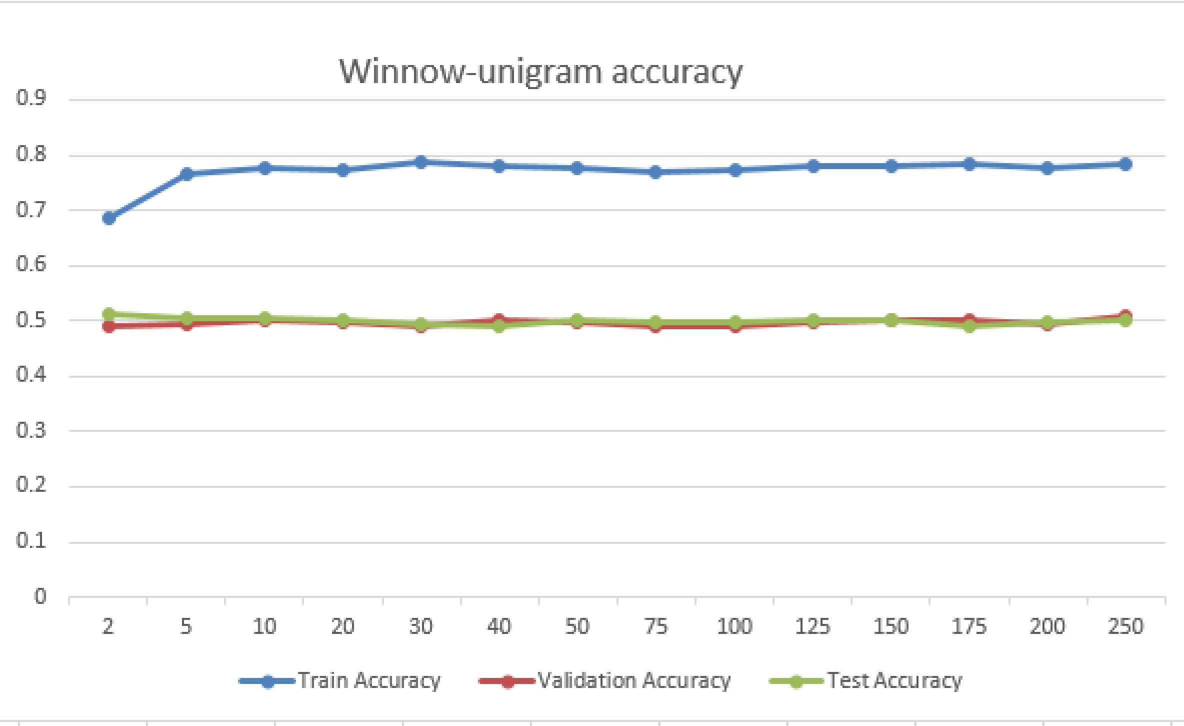
From, the validation accuracy, we find that the best maxIter is 10. The precision,recall and F-score of train data, validation data and test data are in following picture.



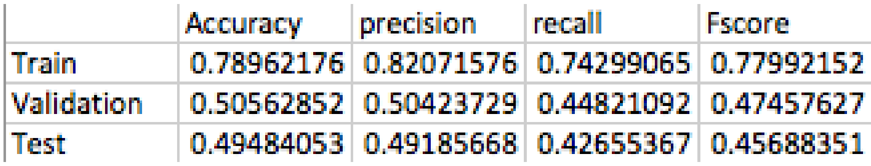
	Accuracy	precision	recall	Fscore
Train	0.99703	0.997816	0.996262	0.997038
Validation	0.701689	0.705323	0.694757	0.7
Test	0.698874	0.693685	0.698955	0.69631

2.4.2 Winnow

The following picture shows the accuracy by using winnow algorithm. The word list is unigram words. We could know that, the accuracy of the train data is relative high when compare with the validation accuracy.

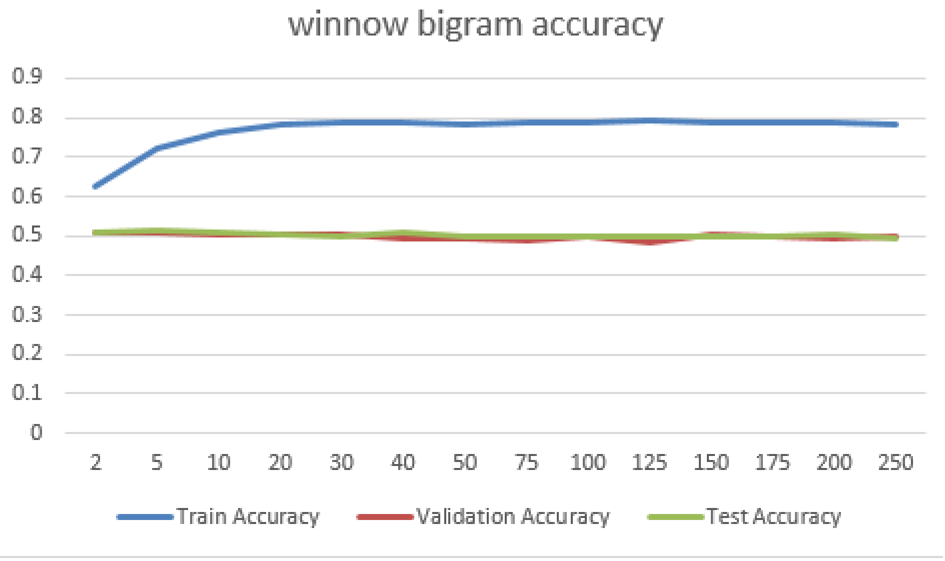


From, the validation accuracy, we find that the best maxIter is 30. The precision,recall and F-score of train data, validation data and test data are as following picture.

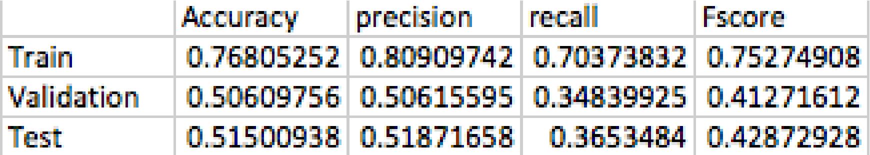


	Accuracy	precision	recall	Fscore
Train	0.78962176	0.82071576	0.74299065	0.77992152
Validation	0.50562852	0.50423729	0.44821092	0.47457627
Test	0.49484053	0.49185668	0.42655367	0.45688351

The following figure shows the accuracy by using winnow algorithm. The word list is bigram words. We could know that, the accuracy of the train data is relative high when compare with the validation accuracy.



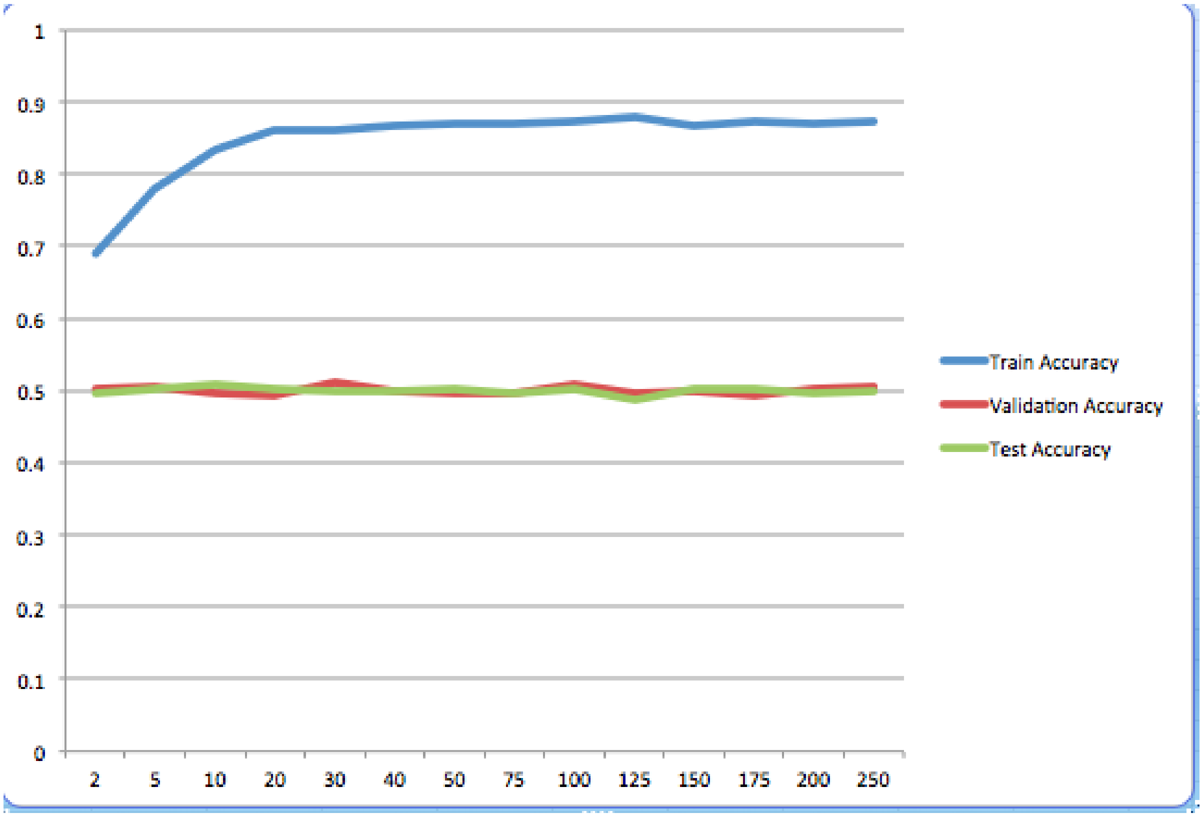
From, the validation accuracy, we find that the best maxIter is 10. The precision,recall and F-score of train data, validation data and test data are as following figure.



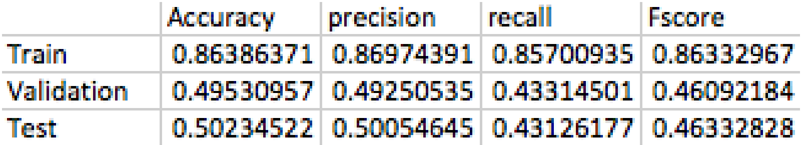
	Accuracy	precision	recall	Fscore
Train	0.76805252	0.80909742	0.70373832	0.75274908
Validation	0.50609756	0.50615595	0.34839925	0.41271612
Test	0.51500938	0.51871658	0.3653484	0.42872928

Compare with unigram, it's performance is not good. The reason could be overfitted. That is one of the reason why it perform not well.

The following figure shows the accuracy by using winnow algorithm. The word list are combination fo unigram and bigram words. We could know that, the accuracy of the train data is relative high when compare with the validation accuracy.



From, the validation accuracy, we find that the best maxIter is 30. The precision,recall and F-score of train data, validation data and test data are as following figure.



	Accuracy	precision	recall	Fscore
Train	0.86386371	0.86974391	0.85700935	0.86332967
Validation	0.49530957	0.49250535	0.43314501	0.46092184
Test	0.50234522	0.50054645	0.43126177	0.46332828

Compare with unigram and bigram, it's performance is relative better. However, the validation and test result are not so good.

2.4.3 Conclusion

From the project, we know, the perceptron algorithm is better than the winnow algorithm. For the perceptron algorithm, it always converge when maxIter become larger. However, it won't affect accuracy the validation data a lot. For the bigram and unigram word list. How many we pick do affect the result. We could not select too much or too little.