COMP 551: Applied Machine Learning
Assignment #2
Report

By:
Weishi Wang (260540022)

Submitted to:
Prof. Sarath Chandar

Oct 21st, 2018

Question 1.

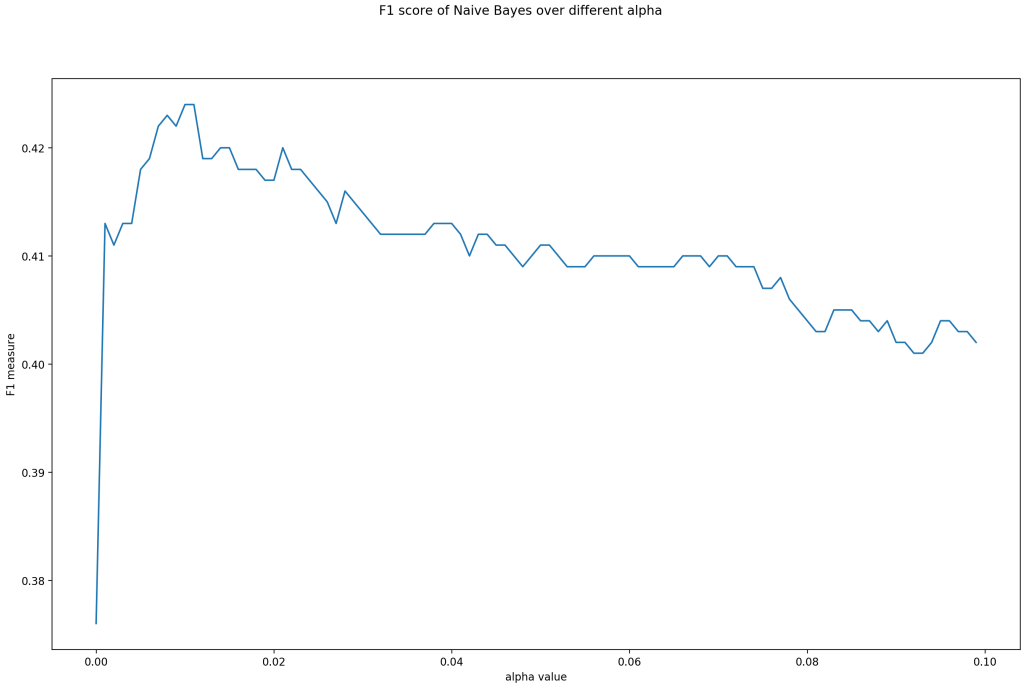
Both IMDB and yelp data are converted into BBoW and FBoW forms. See function "modify","BBoW" and "FBoW".

Question 2.

(a) The performance of random uniform is: 0.1885
The performance of random majority is: 0.351

These data are reported in "Assignment3_260540022_2_a.txt"

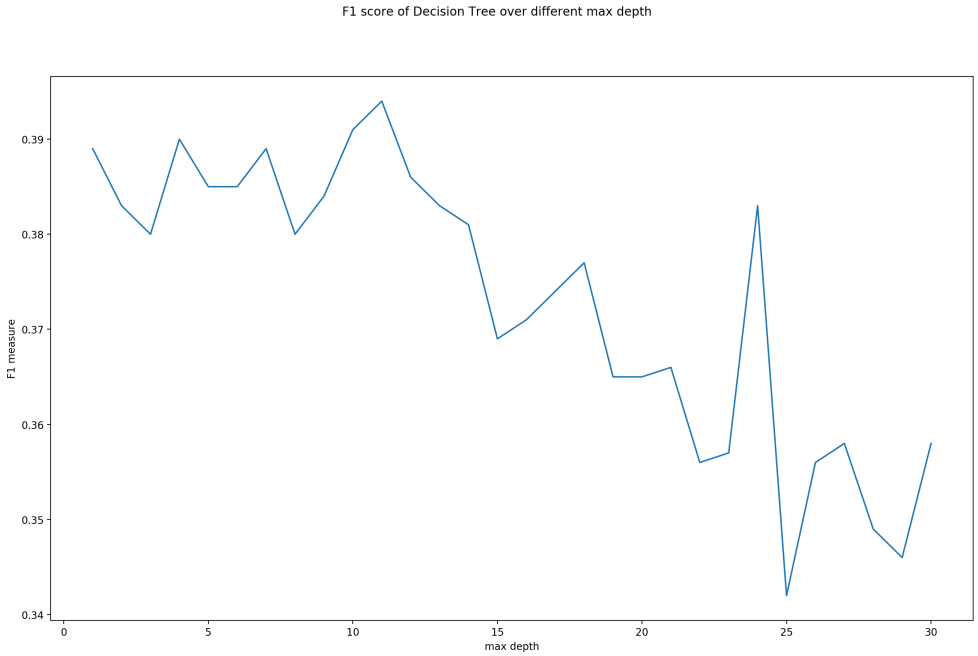
(b)
1. Tune Naïve Bayes:
alpha:



best alpha: 0.0100 (range: 1e-10 to 0.1)

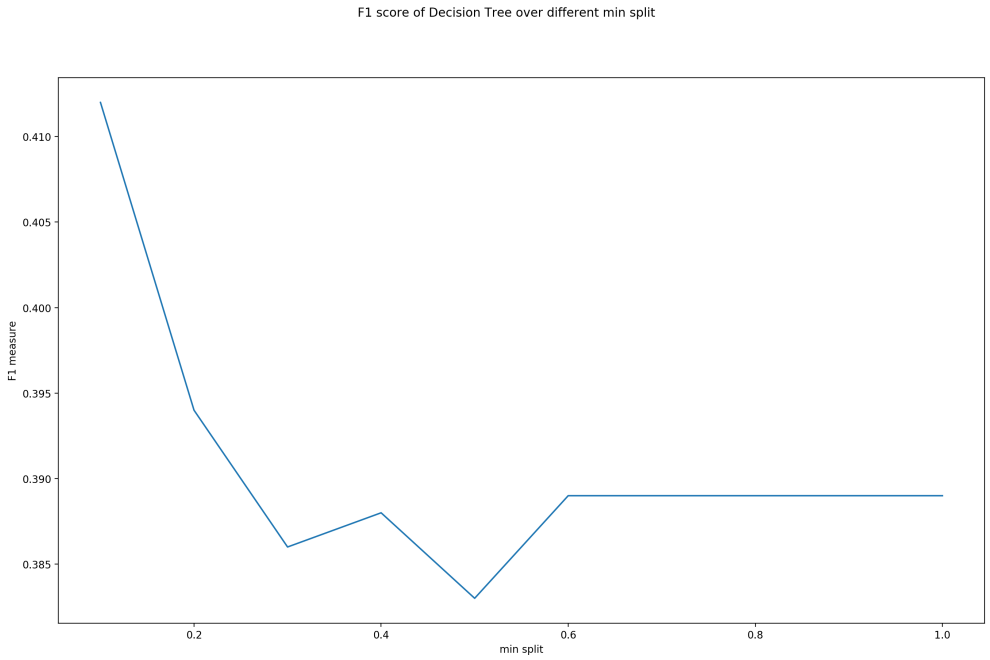
2. Tune Decision Tree:

Depth:



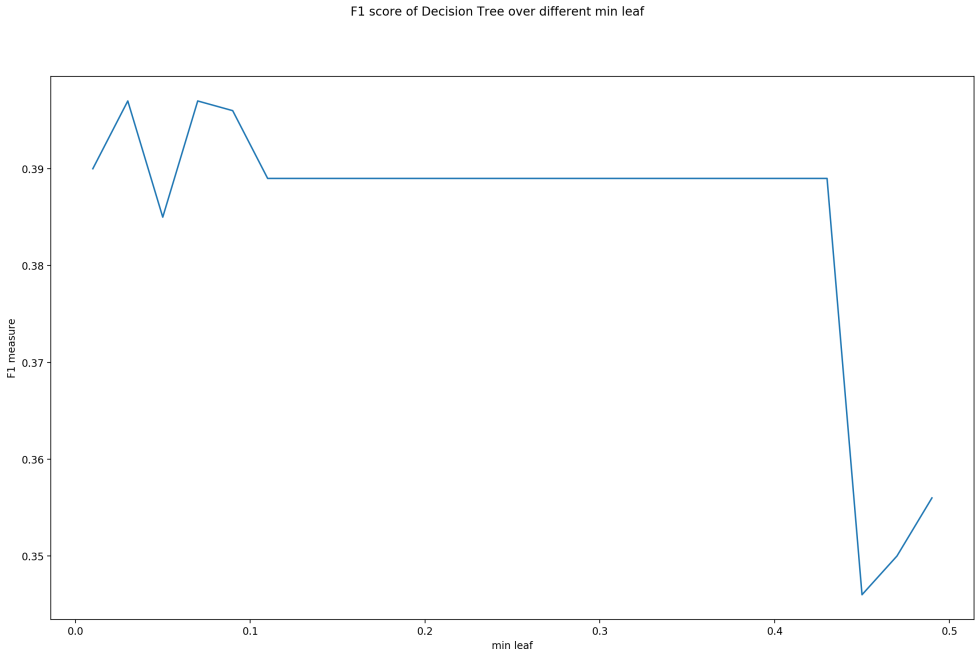
The best max depth: 11 (range :1 to 30)

Min sample split:



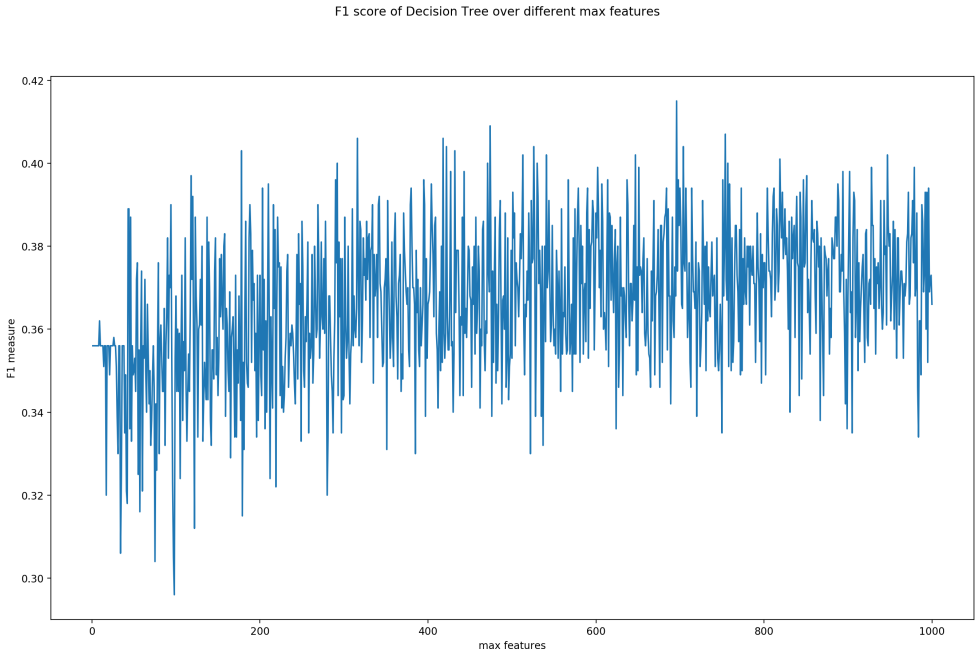
The best min split is: 0.1 (range :0 to 1)

Min sample leaf:



The best min leaf is: 0.03 (range :0 to 0.5)

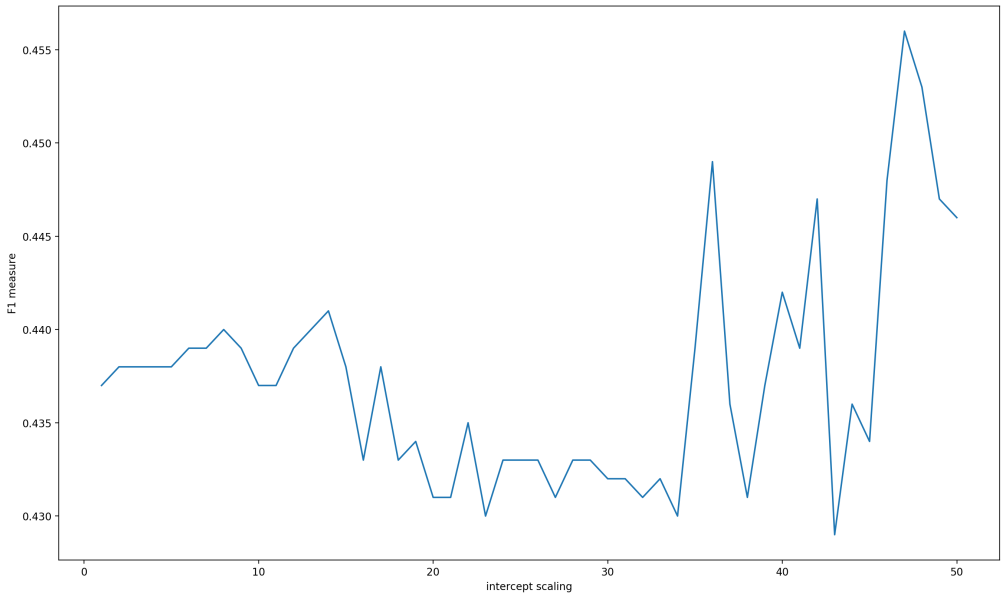
Max features:



The best max features: 696 (range :1 to 1000)

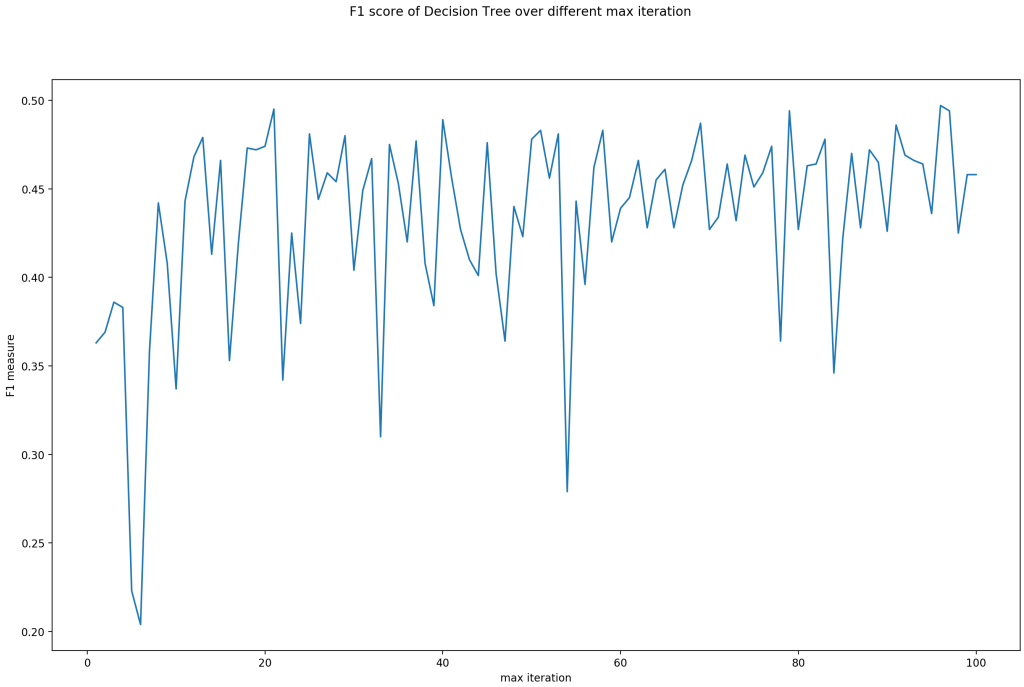
3. Tune linear svm

Intercept scaling:



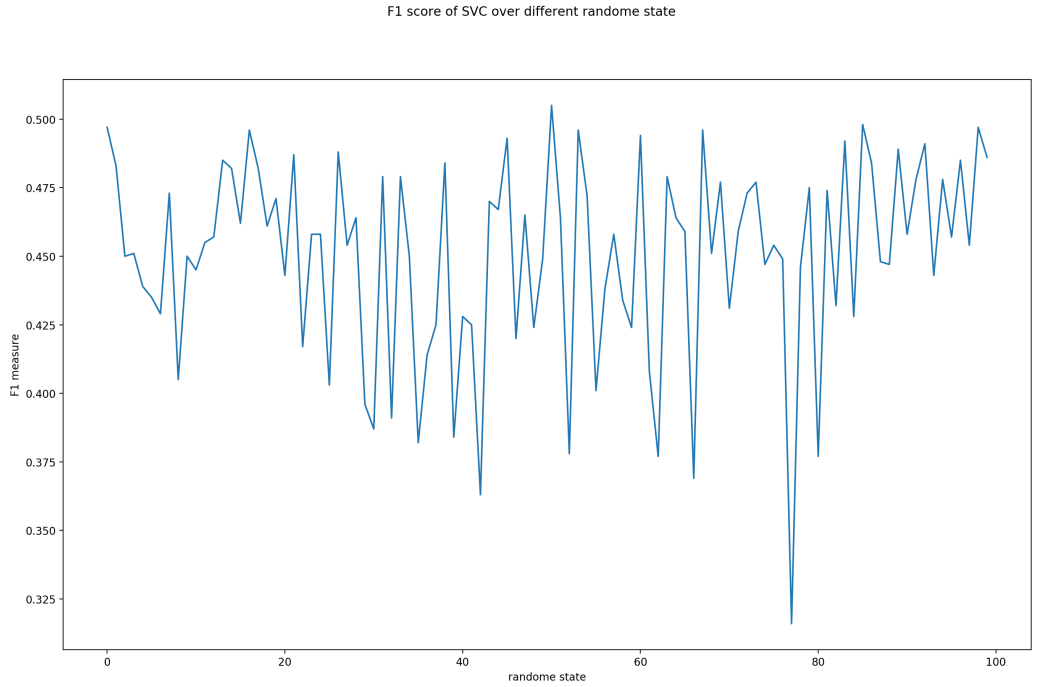
The best intercept scaling: 47 (range :1 to 50)

Max iteration:



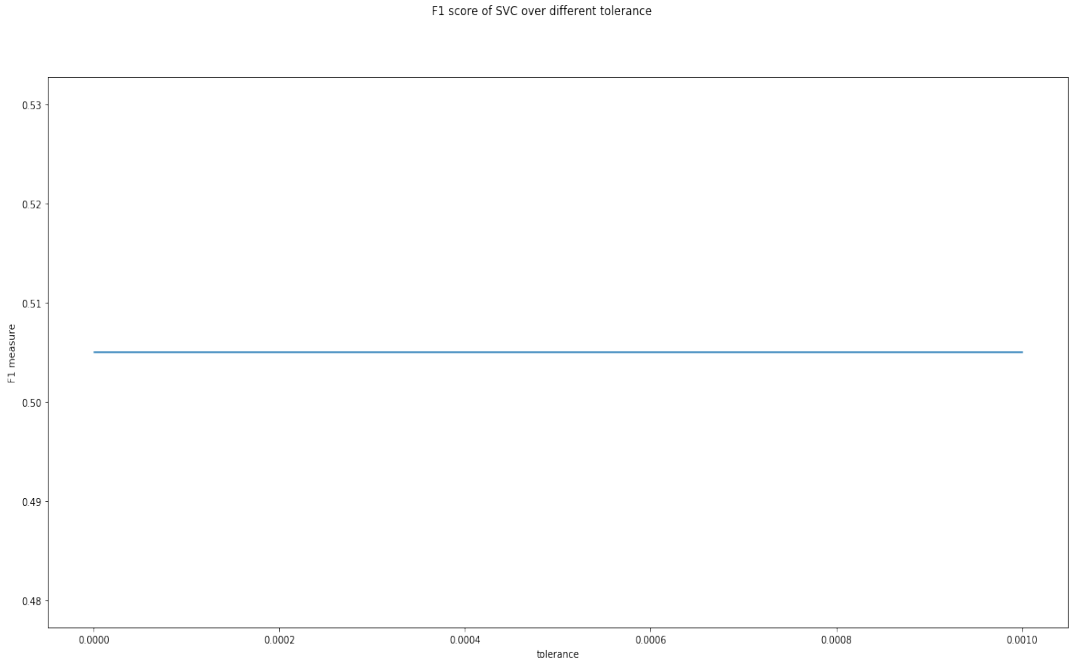
The best max iteration is: 96 (range :1 to 100)

Random state:



The best random state is: 50 (range :1 to 100)

Tolerance:



The best tolerance is: 1e-06 (range :1e-6 to 1e-3)

(c)
Naïve Bayes: best alpha: 0.0100 (range: 1e-10 to 0.1)

Decision Tree:
The best max depth: 11 (range :1 to 30)
The best min split is: 0.1 (range :0 to 1)
The best min leaf is: 0.03 (range :0 to 0.5)
The best max features: 696 (range :1 to 1000)

Linear SVM:
The intercept scaling: 47 (range :1 to 50)
The best max iteration is: 96 (range :1 to 100)
The best random state is: 50 (range :1 to 100)
The best tolerance is: 1e-06 (range :1e-6 to 1e-3)

These data are reported in files
"Assignment3_260540022_2_c_naive_bayes_hyperparameter.txt",
"Assignment3_260540022_2_c_decision_tree_hyperparameter.txt" and
"Assignment3_260540022_2_c_svm_hyperparameter.txt"

(d)
Train performance:
naive bayes: 0.7305714285714285
decision tree: 0.41228571428571426
svm: 0.898857142857143

Valid performance:
naive bayes: 0.424
decision tree: 0.377
svm: 0.505

Test performance:
naive bayes: 0.444
decision tree: 0.396
svm: 0.501

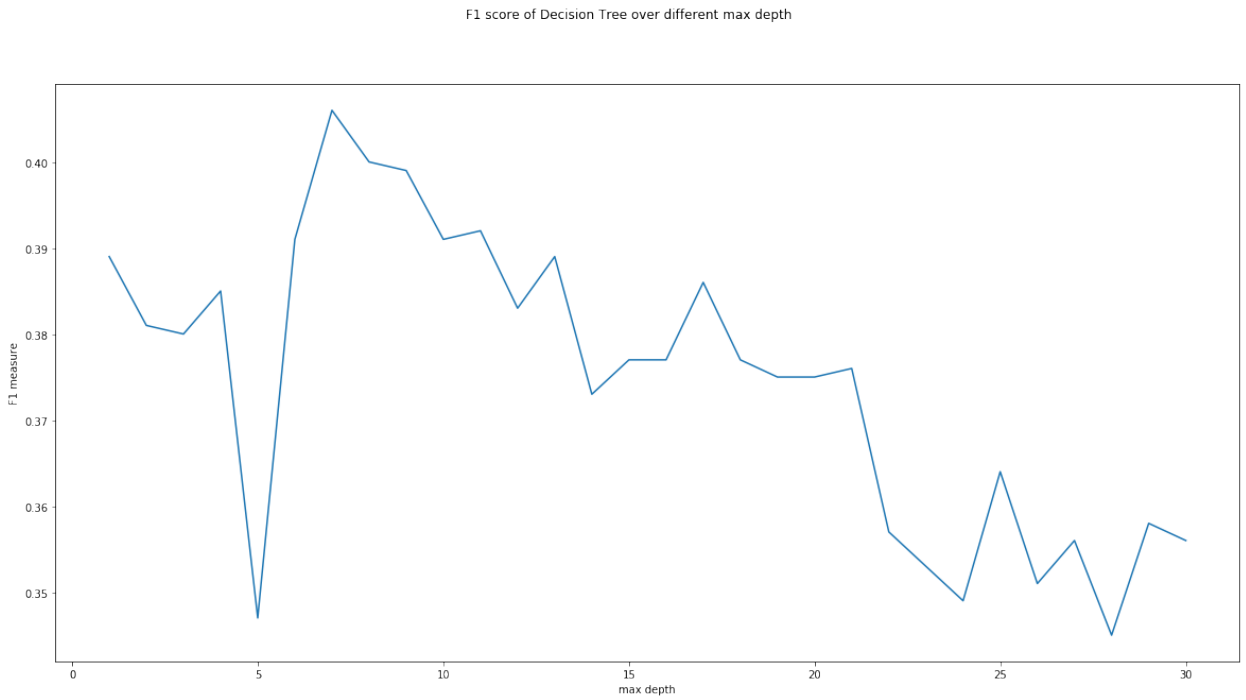
These data are reported in file "Assignment_260540022_2_d.txt"

(e)
For this data set, the linear support vector machine performs slightly better than other models. It has a F1 score of 0.501, which is not bad for a 5 classes classification, and given that it is higher than the random classifier. The reason for SVM classifier to have a higher performance is that the data might be near linearly separable, and high dimensionality. SVM will simply place a hyperplane and maximize the margin to both classes, so that higher dimension does not affect the classification too much. However, naïve Bayes and decision tree tends to have poor performance on high dimensional data set.

Question 3.

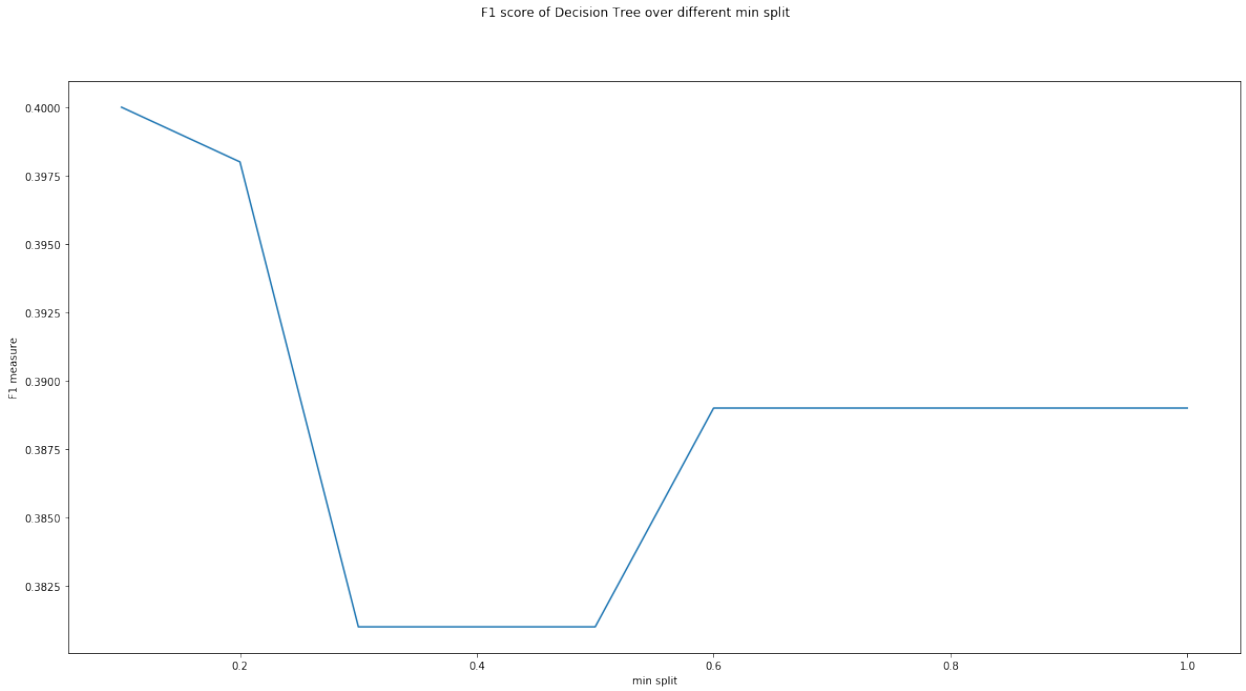
(a)
1. Gaussian Naïve Bayes will not be tuned

2. Tune Decision Tree:
depth:



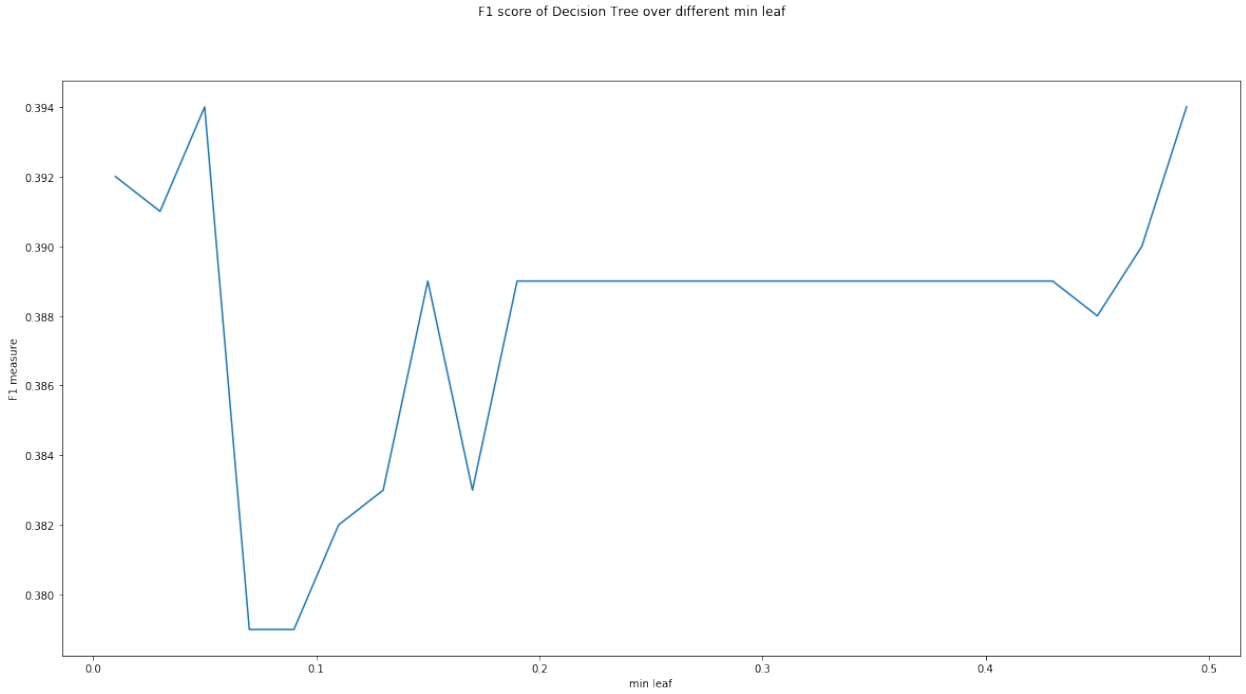
The best max depth: 7 (range :1 to 30)

Min sample split:



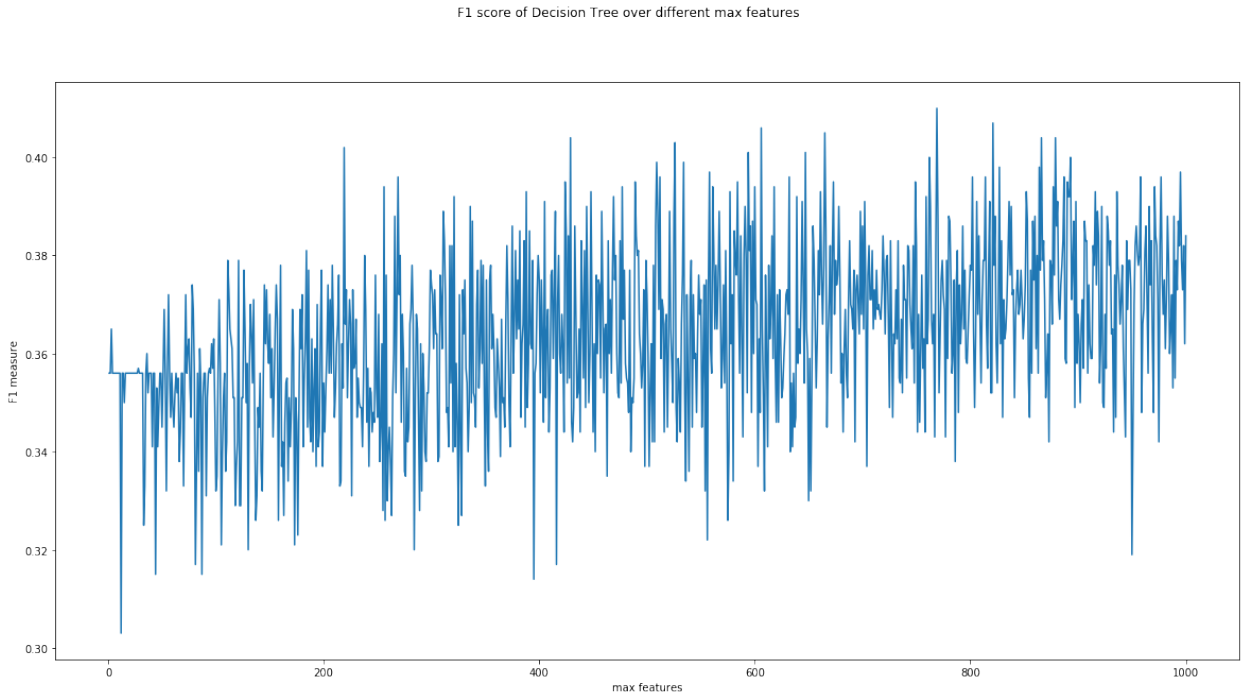
The best min split is: 0.1 (range :0 to 1)

Min sample leaf:



The best min leaf is: 0.05 (range :0 to 0.5)

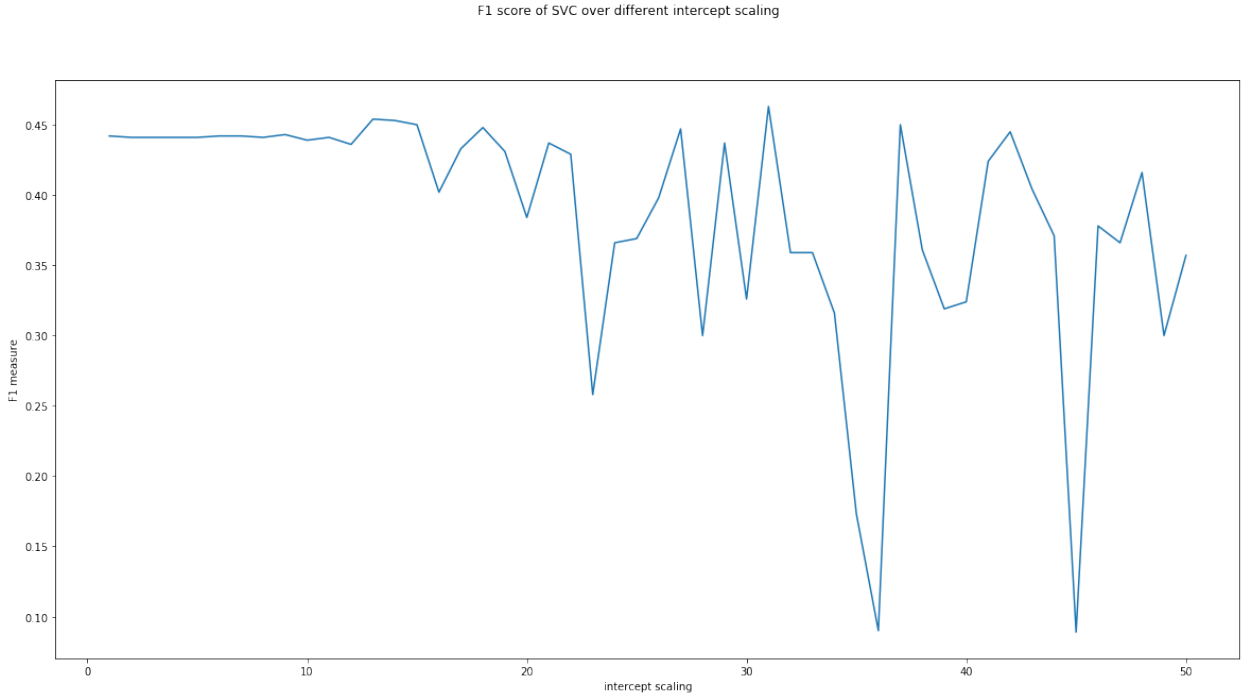
Max features:



The best max features: 769 (range :1 to 1000)

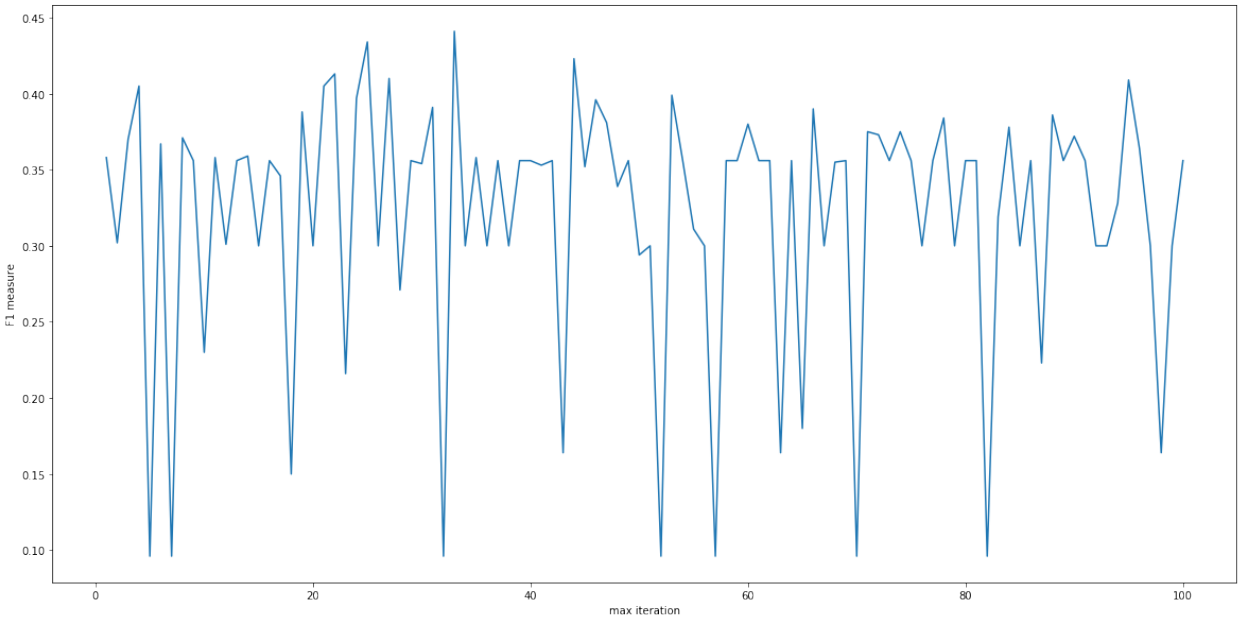
3. Tune linear svm

Intercept scaling:



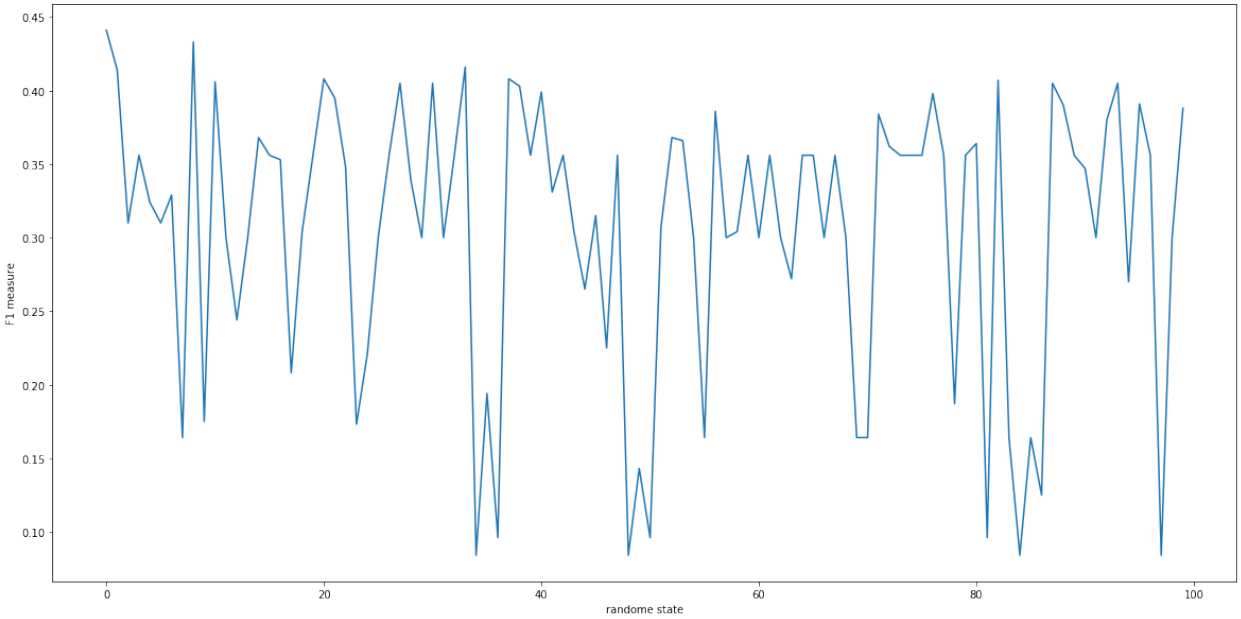
The best intercept scaling: 31 (range: 1 to 50)

Max iteration:



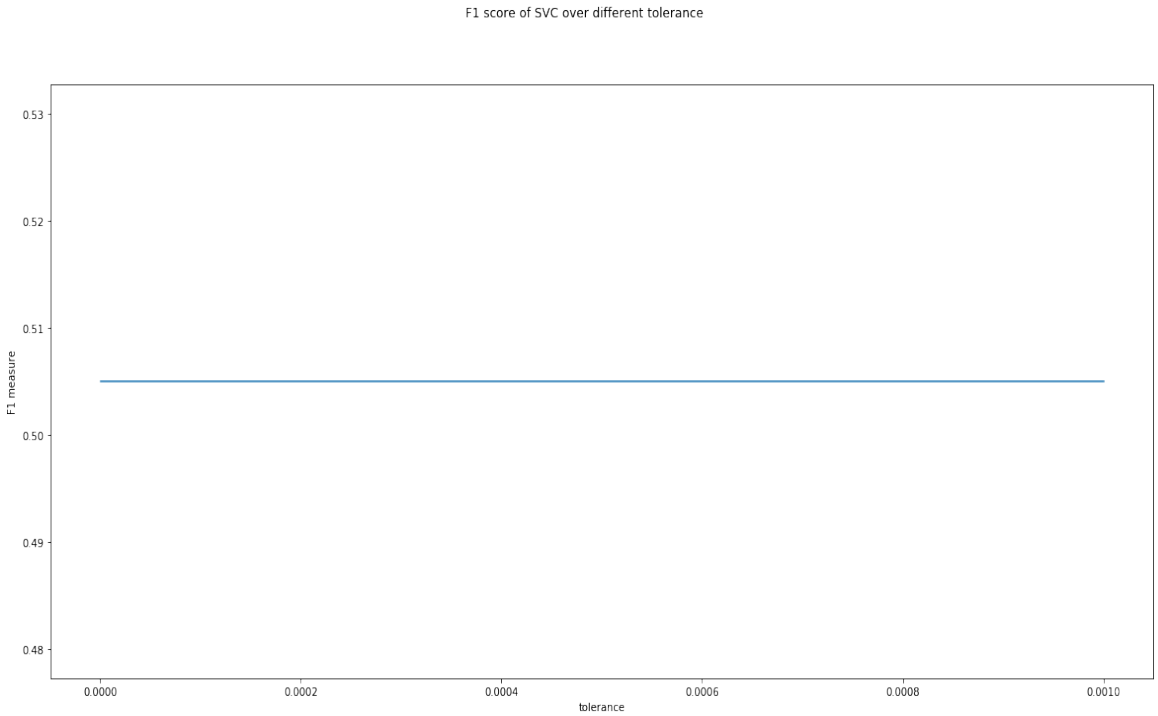
The best max iteration: 33 (range :1 to 100)

Random state:



The best max random state: 0 (range :1 to 100)

Tolerance:



The best max tolerance: 1e-06 (range : 1e-6 to 1e-3)

(b)

Decision Tree:

The best max depth: 7 (range :1 to 30)
The best min split is: 0.1 (range :0 to 1)
The best min leaf is: 0.05 (range :0 to 0.5)
The best max features: 769 (range :1 to 1000)

SVM:

The best intercept scaling: 31 (range :1 to 50)
The best max iteration: 33 (range :1 to 100)
The best max randome state: 0 (range :1 to 100)
The best max tolerance: 1e-06 (range :1e-6 to 1e-3)

These data are stored in "Assignment_260540022_3_b_decision_tree.txt" and "Assignment_260540022_3_b_svm.txt"

(c)
Train performance:
naive bayes: 0.7937142857142857
decision tree: 0.399
svm: 0.44385714285714284

Valid performance:
naive bayes: 0.301
decision tree: 0.37
svm: 0.441

Test performance:
naive bayes: 0.32
decision tree: 0.369
svm: 0.425

These data are reported in file "Assignment_260540022_3_c.txt"

(d)
The SVM still performs the best, in the FBoW representation. This is mainly due to the high dimensionality of data set and its natural near linearly separable distribution. According to the graphs generated for different hyper parameters in SVM, the most important hyper is intercept scaling, which is best at 31. This hyper parameter is not chaotic like others, and actually have a trend. It helps to optimize margin and stabilize the performance.

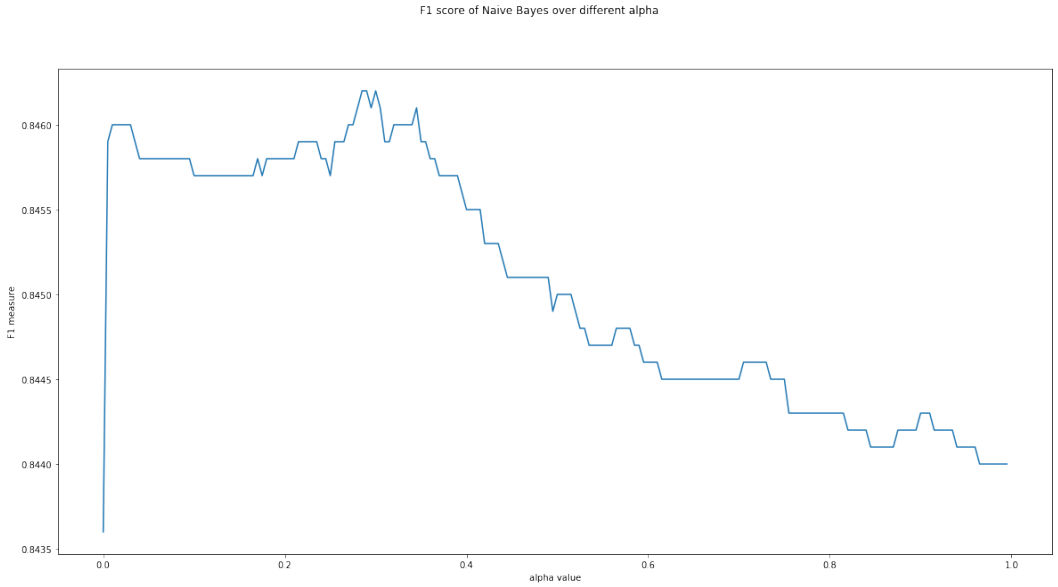
(e)
According to the performances of both data representation, the BBoW generally performs better than FBoW. The main difference between these two representations is the occurrence of the words. BBoW does not consider the weight of some words occurring multiple times, whereas FBoW does take it into account. The performance of naïve Bayes is worse than BBoW, this is likely to be cause by some high frequent words over shadowing other words which probably have important meaning. This is likely to enhance frequent words and reduce importance of less frequent words. In a data consists of "reviews", there are some highly repetitive words such as "I", "and", "a" and "the" etc. Therefore, the performances for decision tree and also linear support vector machine also decrease.

(f)
This depends on the scale of the data and the structure of the data. If a data is very large or very repetitive, FBoW will have many repeated words which have little meaning and will shadow other useful words. In this case, BBoW will perform better. However, if the data is small and non-repetitive, FBoW can represent the importance of certain words while not over shadow some less repetitive but equally important words. For this data set, the BBoW performs better since there are many repetitive words.

Question 4.

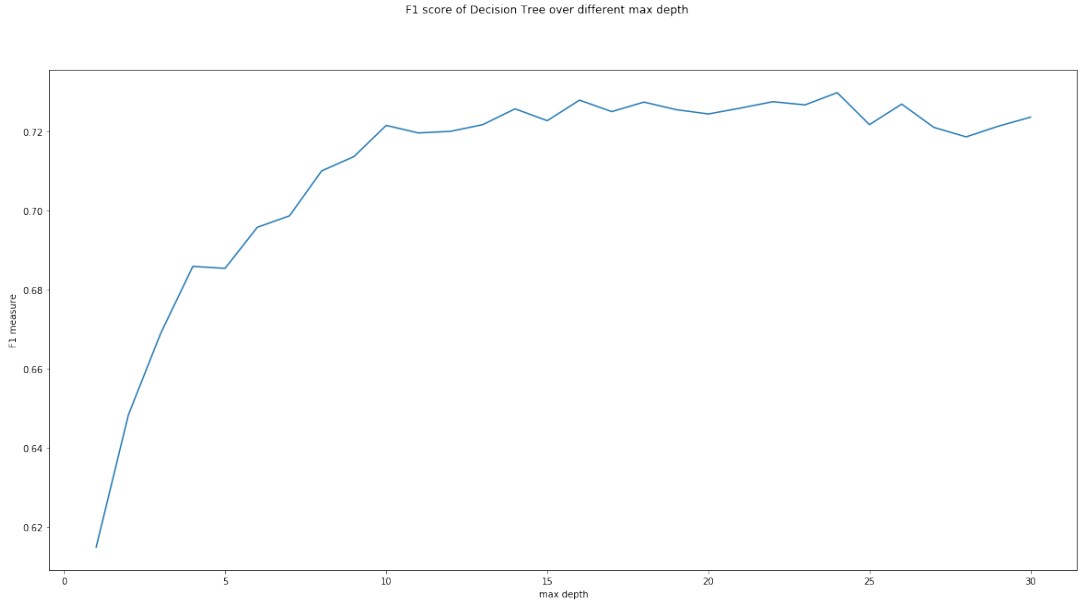
(a)
Random uniform IMDB uniform: 0.50056

(b)
1. Tune Naïve Bayes:
alpha:



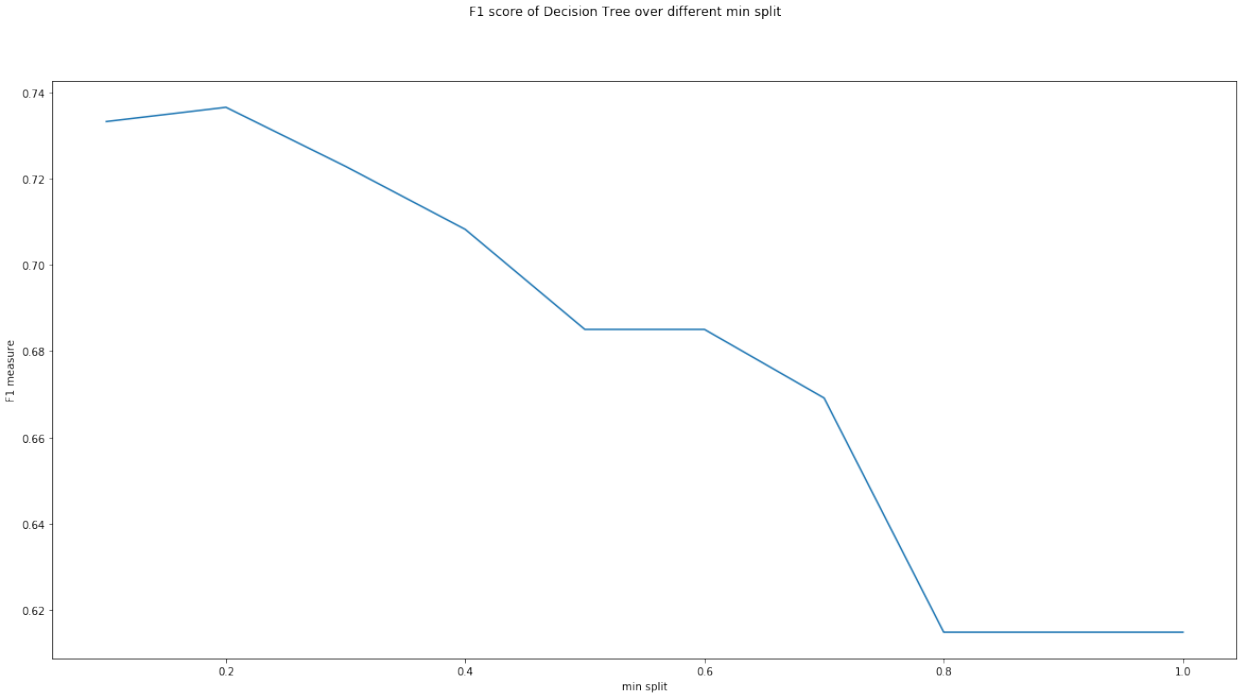
best alpha: 0.285 (range: 1e-10 to 1)

2. Tune Decision Tree:
Depth:



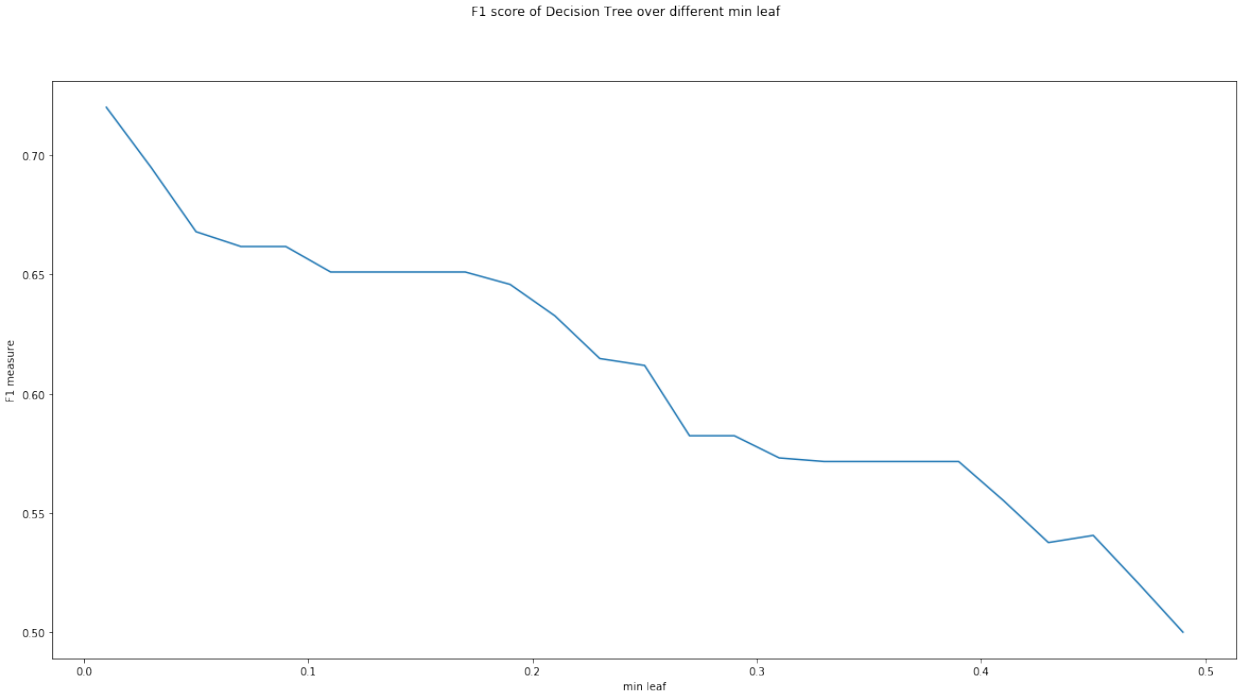
The best max depth: 24 (range :1 to 30)

Min sample split:



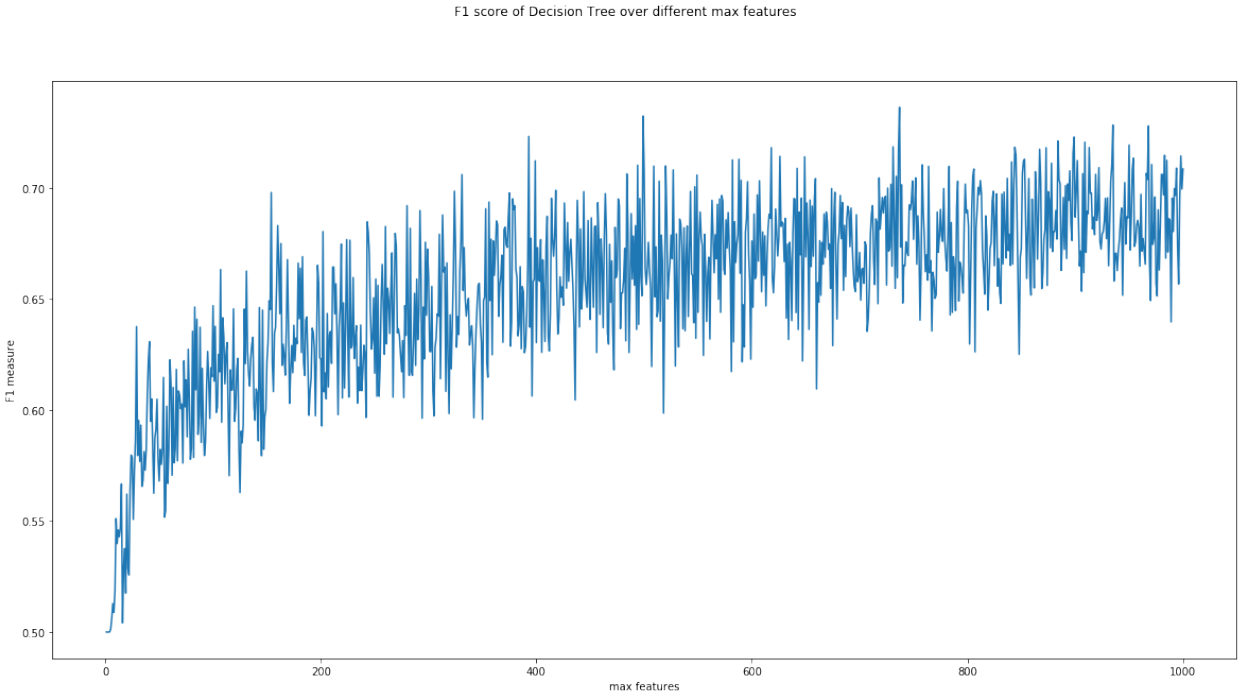
The best min split is: 0.2 (range :0 to 1)

Min sample leaf:



The best min leaf is: 0.01 (range :0 to 0.5)

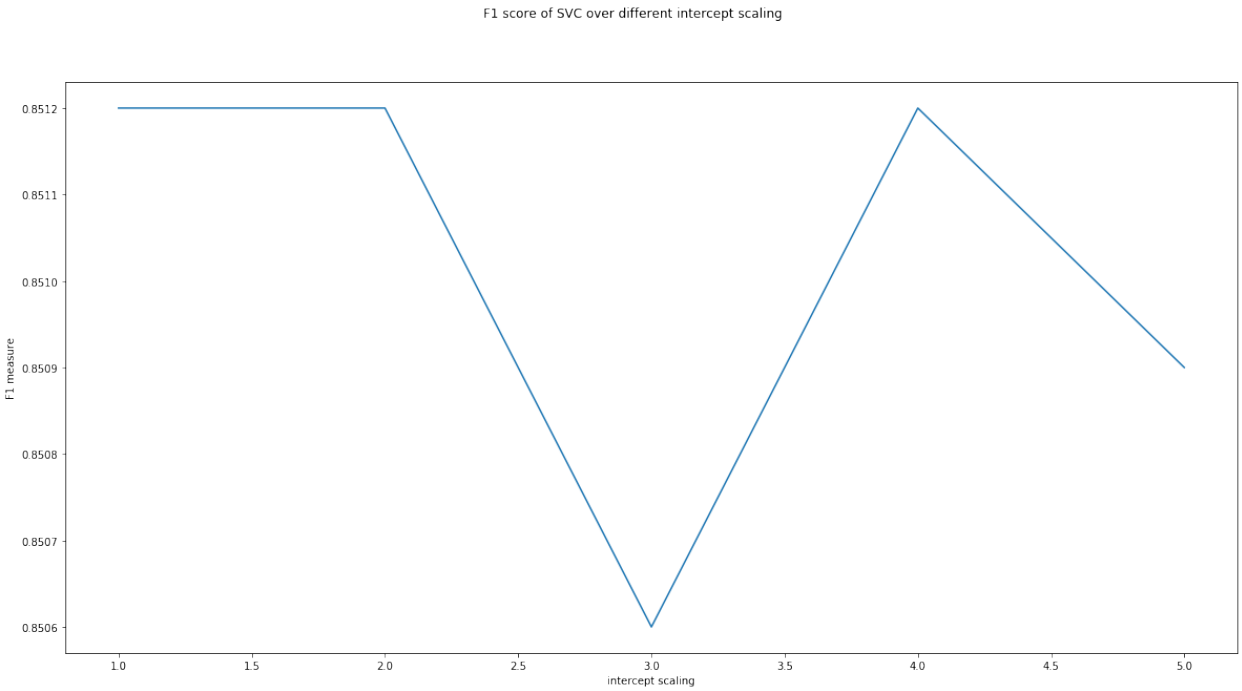
Max features:



The best max features: 737 (range :1 to 1000)

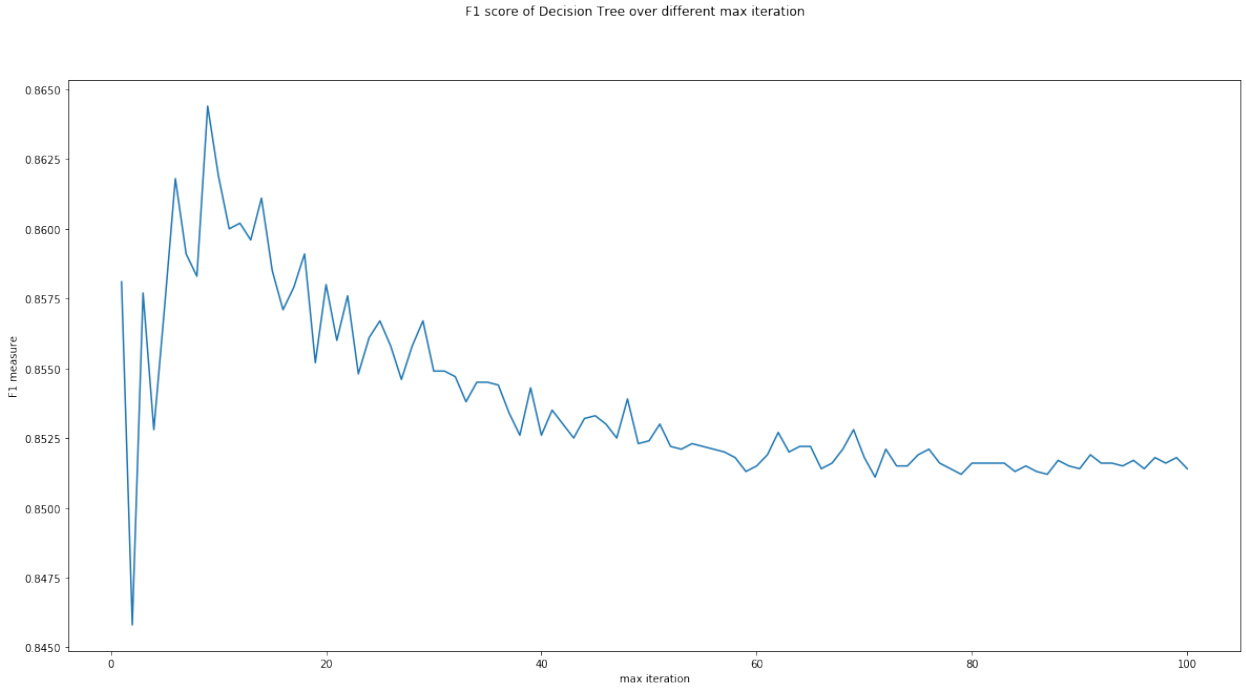
3. Tune linear svm

Intercept scaling:



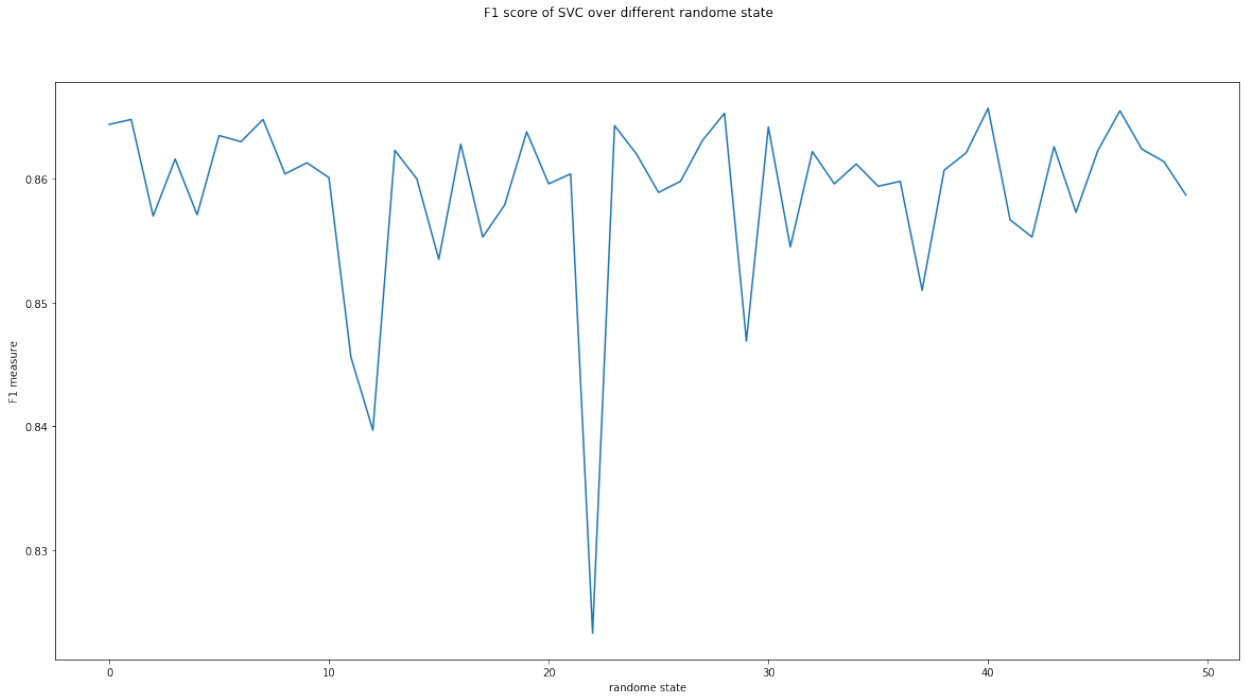
The best intercept scaling: 1 (range :1 to 50)

Max iteration:



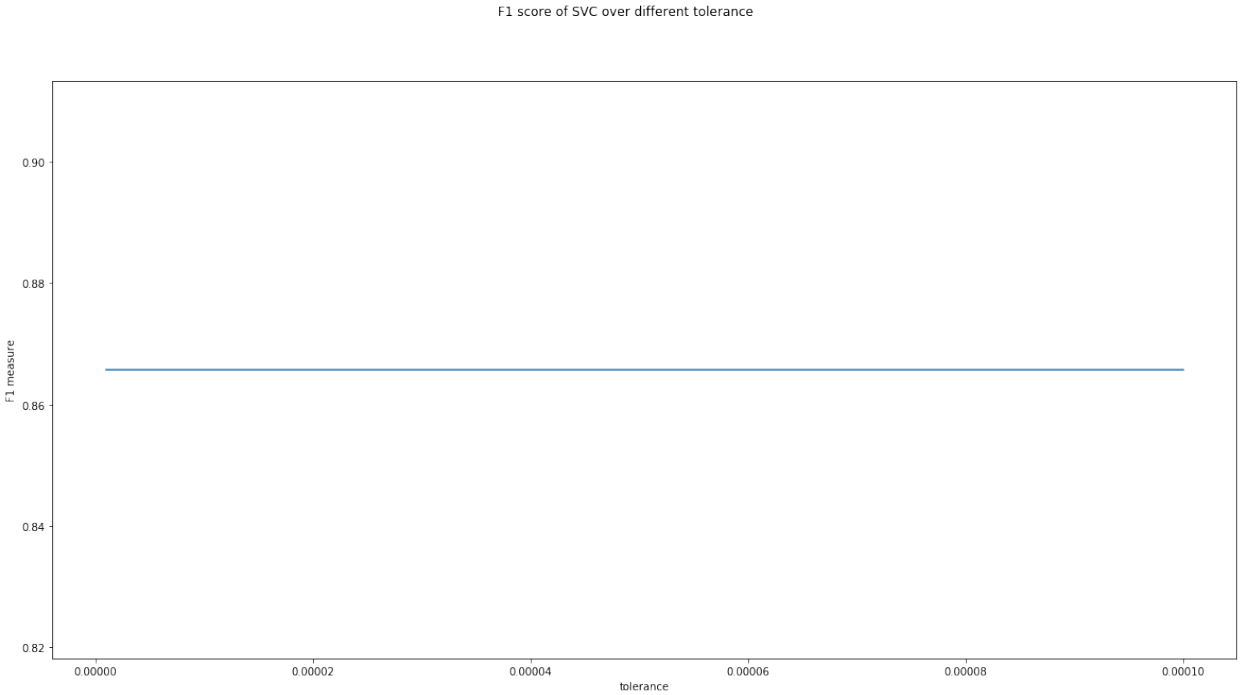
The best max iteration is: 9 (range :1 to 100)

Random state:



The best random state is: 40 (range :1 to 100)

Tolerance:



The best tolerance is: 1e-06 (range :1e-6 to 1e-3)

(c)
Naïve Bayes:
best alpha: 0.28500000010000015 (range: 1e-10 to 1)

Decision Tree:
The best max depth: 24 (range :1 to 30)
The best min split is: 0.2 (range :0 to 1)
The best min leaf is: 0.01 (range :0 to 0.5)
The best max features: 737 (range :1 to 1000)

SVM:
The intercept scaling: 1 (range :1 to 50)
The best max iteration is: 9 (range :1 to 100)
The best randome state is: 40 (range :1 to 100)
The best tolerance is: 1e-06 (range :1e-6 to 1e-3)

These data are stored in “Assignment_260540022_4_c_naive_bayes.txt”, “Assignment_260540022_4_c_decision_tree.txt” and “Assignment_260540022_4_c_svm.txt”

(d)
Train performance:
naive bayes: 0.8723333333333333
decision tree: 0.7065333333333333
svm: 0.9871333333333333

Valid performance:
naive bayes: 0.8485
decision tree: 0.681
svm: 0.8657

Test performance:
naive bayes: 0.83632
decision tree: 0.6728
svm: 0.85144

These data are reported in file “Assignment_260540022_4_d.txt”

(e)
In the IMDB data set, SVM still performs better than other two classifiers but naïve Bayes has very similar performance. Despite the fact that the data are high dimensional, naïve Bayes having good performance indicates that the data are likely to be less redundant, and the data size is also large enough to train naïve Bayes classifier. The linear SVM still performs better due to the high dimensionality.

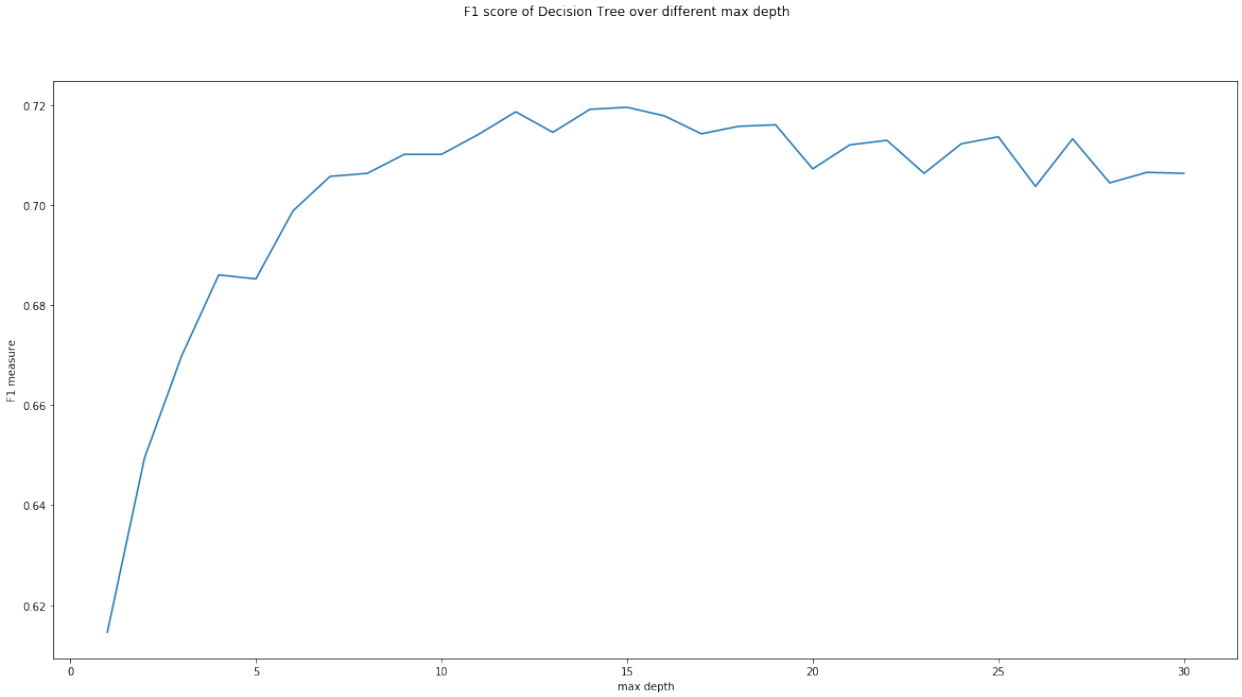
Question 5.

(a)

1. Gaussian Naïve Bayes will not be tuned

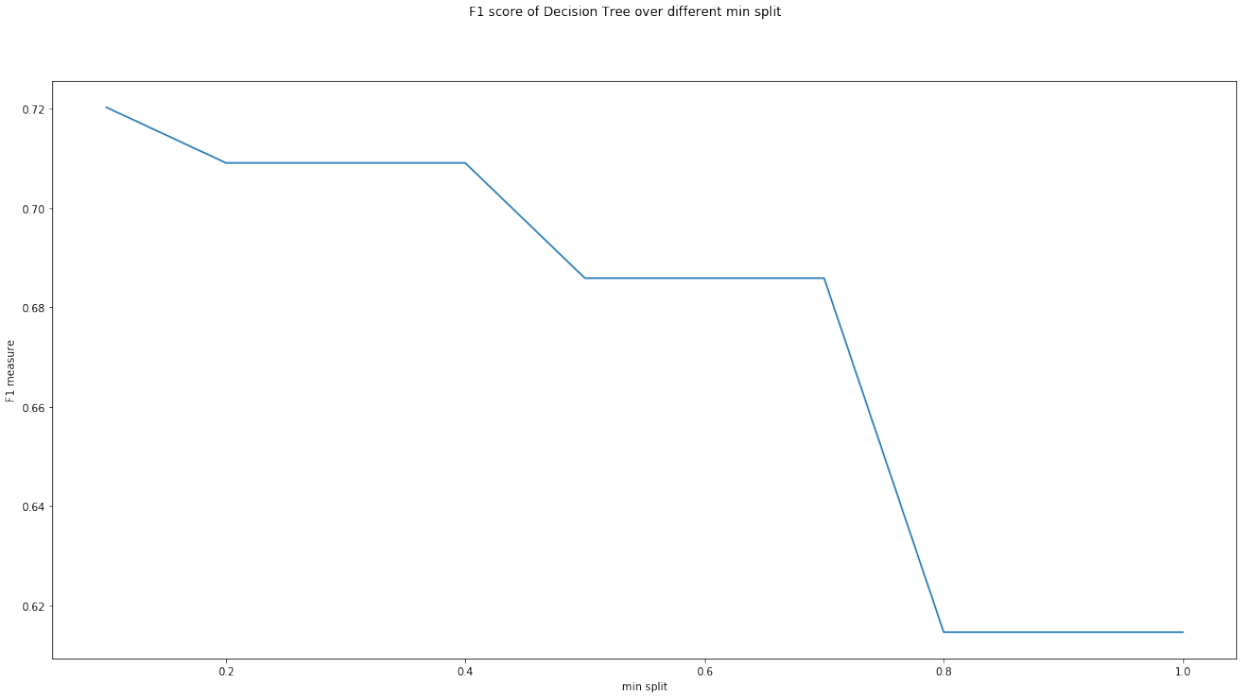
2. Tune Decision Tree:

Depth:



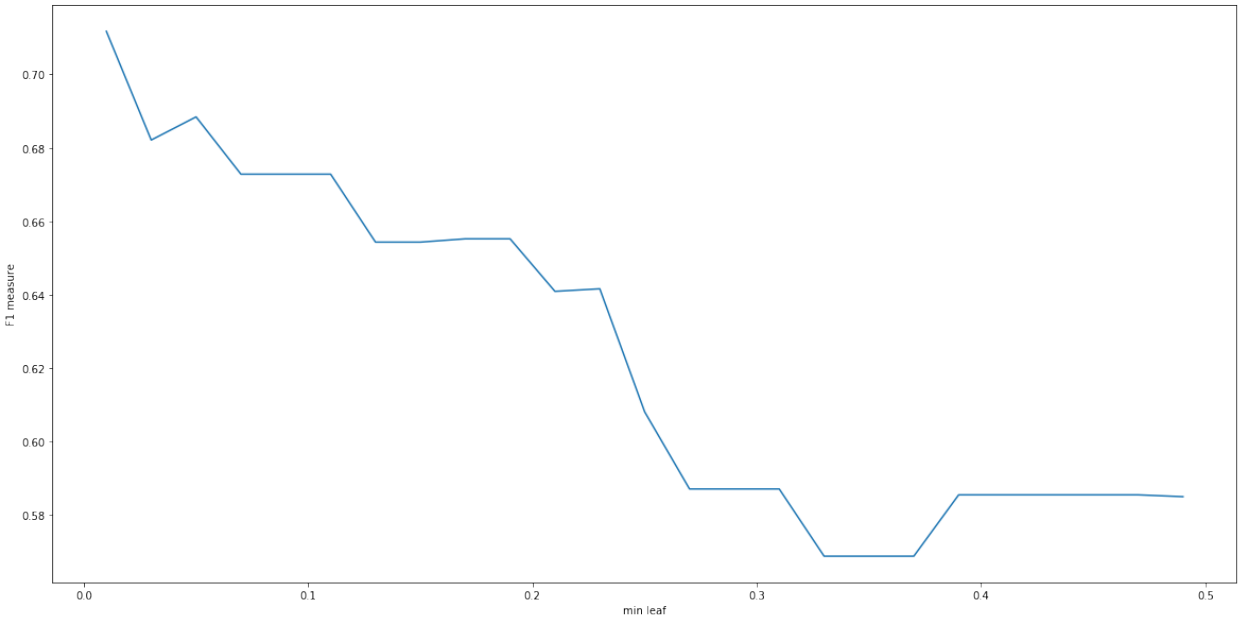
The best max depth: 15 (range :1 to 30)

Min sample split:



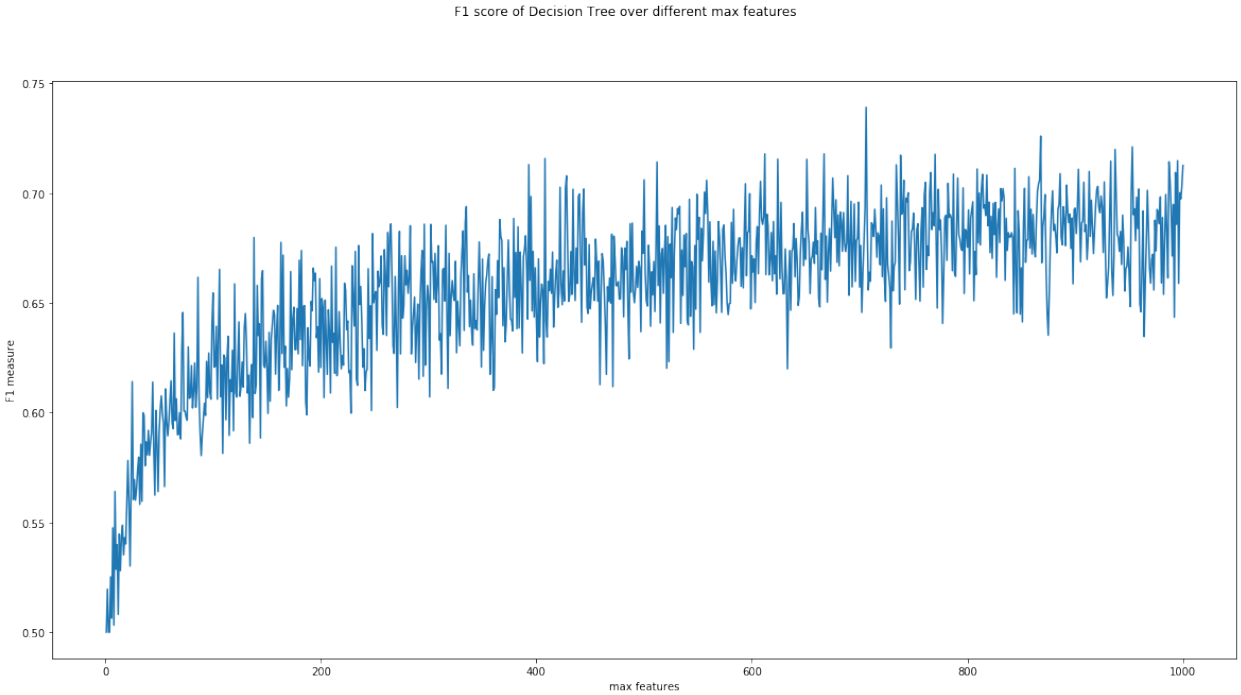
The best min split is: 0.1 (range :0 to 1)

Min sample leaf:



The best min leaf is: 0.01 (range :0 to 0.5)

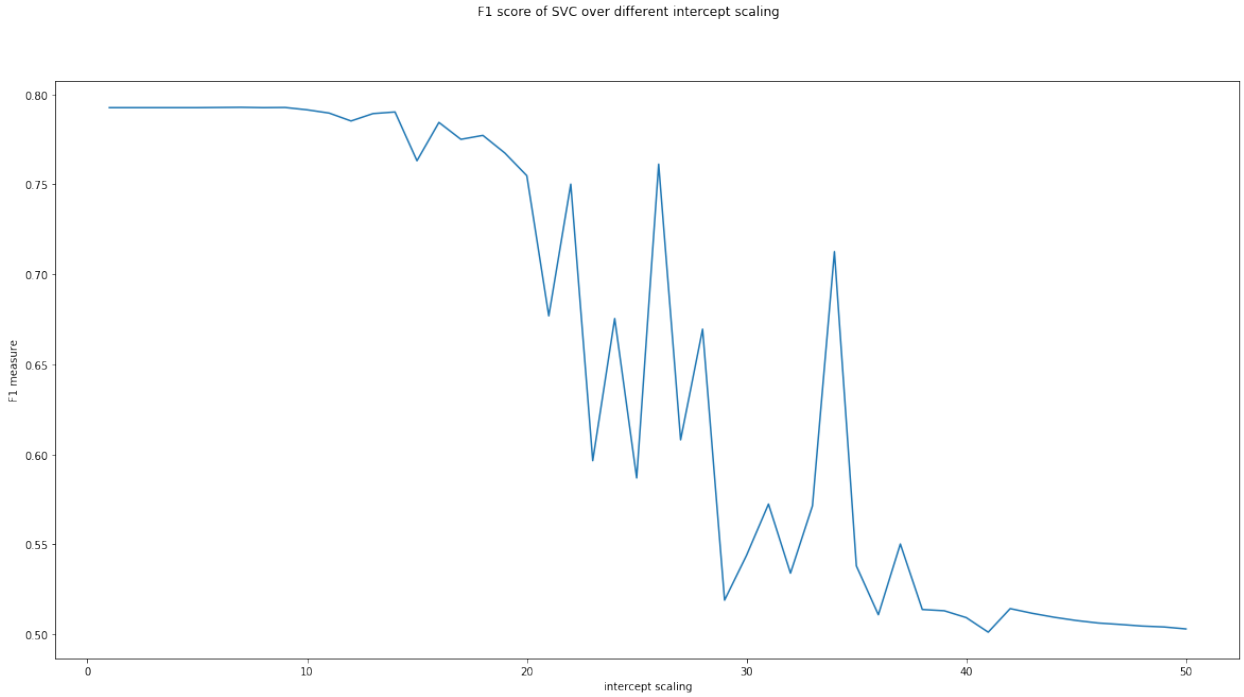
Max features:



The best max features: 706 (range :1 to 1000)

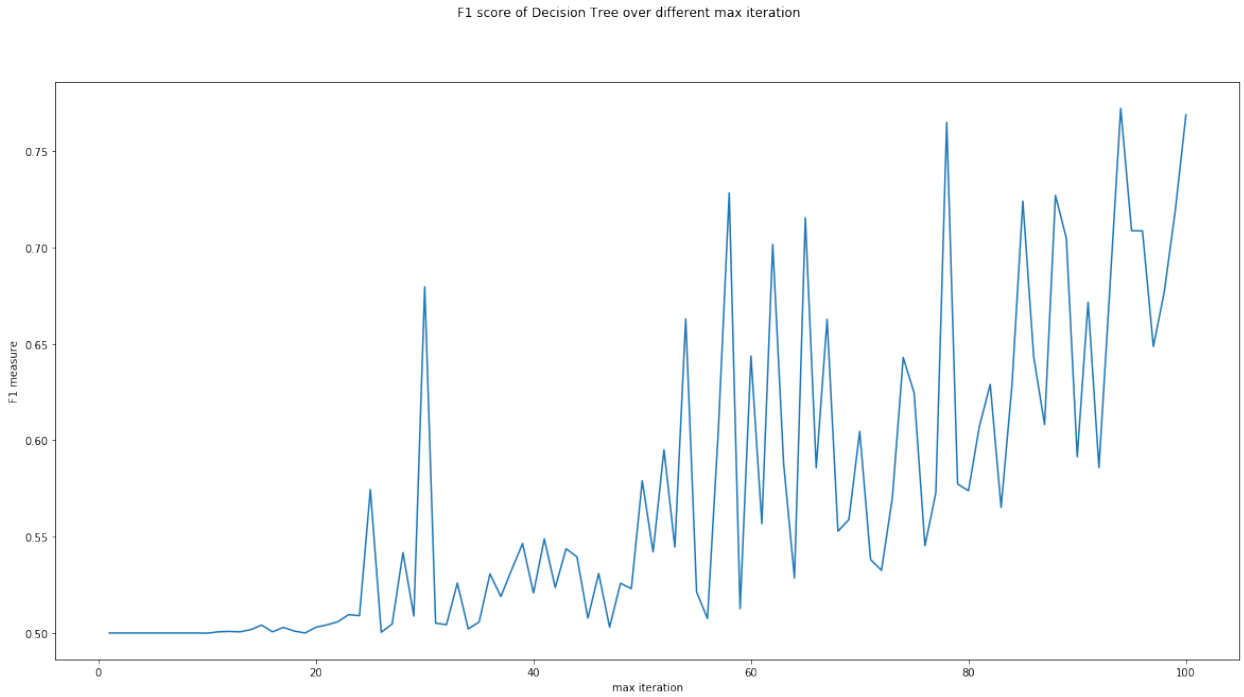
3. Tune linear svm

Intercept scaling:



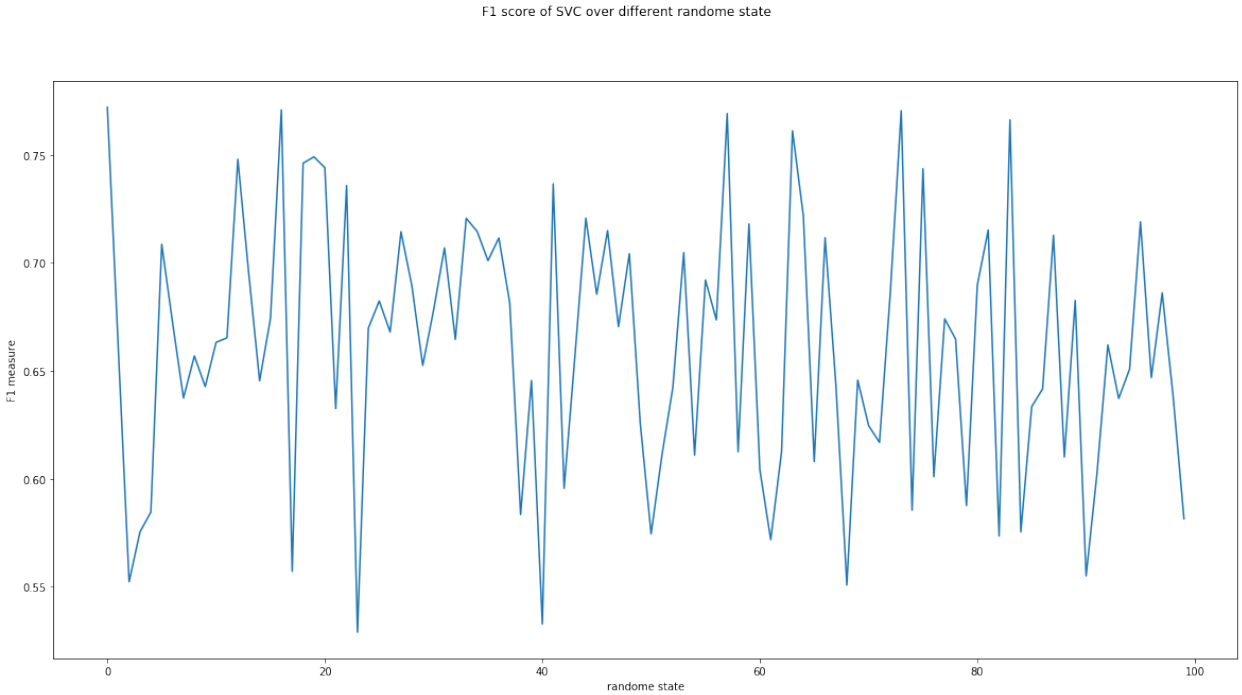
The best intercept scaling: 7 (range: 1 to 50)

Max iteration:



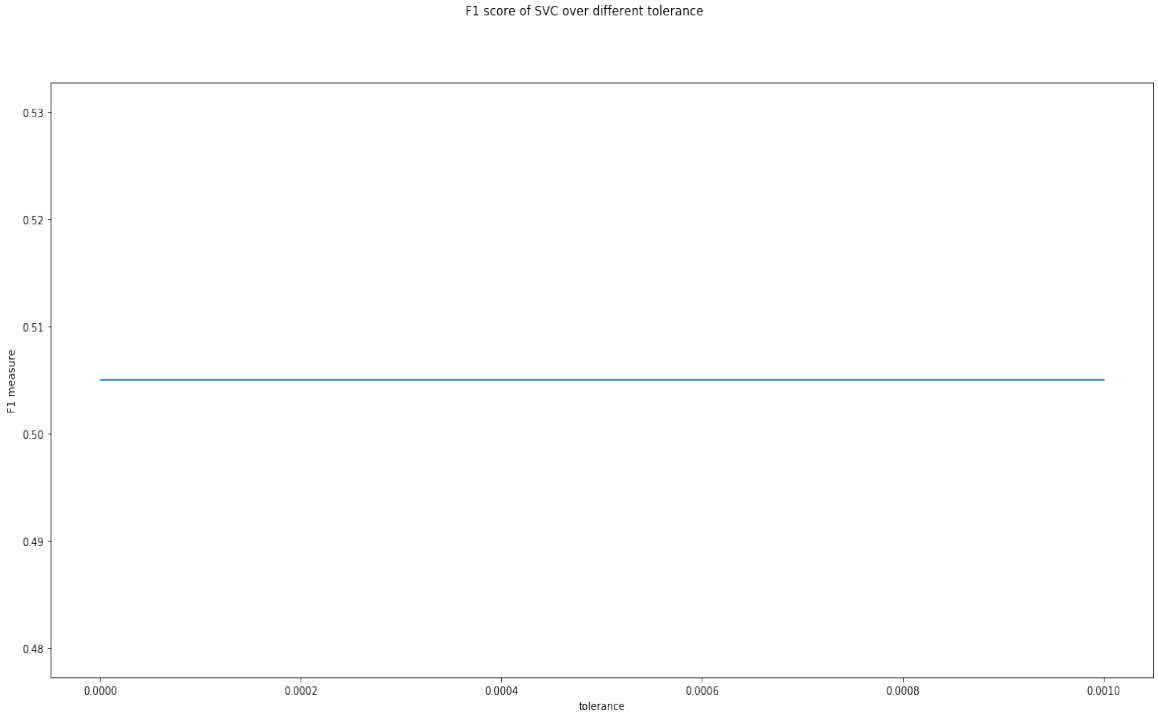
The best max iteration: 94 (range :1 to 100)

Random state:



The best max random state: 0 (range :1 to 100)

Tolerance:



The best max tolerance: 1e-06 (range: 1e-6 to 1e-3)

(b)
Decision Tree:
The best max depth: 15 (range :1 to 30)
The best min split is: 0.1 (range :0 to 1)
The best min leaf is: 0.01 (range :0 to 0.5)
The best max features: 706 (range :1 to 1000)

SVM:
The intercept scaling: 7 (range :1 to 50)
The best max iteration is: 94 (range :1 to 100)
The best random state is: 0 (range :1 to 100)
The best tolerance is: 1e-06 (range :1e-6 to 1e-3)

These data are stored in "Assignment_260540022_5_b_decision_tree.txt" and "Assignment_260540022_5_b_svm.txt"

(c)
Train performance:
naive bayes: 0.8575333333333334
decision tree: 0.6713333333333333
svm: 0.7839333333333334

Valid performance:
naive bayes: 0.7545000000000001
decision tree: 0.6747
svm: 0.7722000000000001

Test performance:
naive bayes: 0.6874
decision tree: 0.66824
svm: 0.7729999999999999

(d)
The model which has the best performance is linear SVM. Same reason to previous question applies here as well. SVM works better than naïve Bayes and decision tree in higher dimension, and according to the relatively higher performance on BBoW, we can also conclude that the data are evenly distributed and near linearly separable. The hyper parameter that helped the model to increase performance is the maximum iteration.

(e)
Similar to previous answer, the performance of FBoW is genrally poorer than BBoW representation. The reason is also because of some frequent words reduce the importance of other less frequent buy very important words. According to my performance data, naïve Bayes, decision tree and SVM are all performing poorer than BBoW representation.

(f)

Same as the question 3(f), the goodness of each representation depends on specific data set features, including size of the data and structure of the data. For this data set, the BBoW performs better than FBoW, since movie reviews will containg many repetitive words and will reduce the importance of other words.

(g)

The performance of different classifiers changes when data set changes, for example, the naïve Bayes classifier works much well on the IMDB data set and yelp data set. This is because the natural distribution of the data. IMDB is more likely to be less redundant and well segmented. The data size of IMDB is also larger than yelp data set, therefore it is more suitable to train naïve Bayes classifier. Moreover, IMDB is only a classes data set, so the performance is expected to be much better than yelp data set, which is a 5 classes data.