let us tell you all about our project

at the beginning we struggled with the best idea , first we thought about starting and building histograms for each celebrity, so that we will have a dictionary for each celeb.

That way we could categorized each twitt to the best match of words from one of those dictionaries.

after giving it a thought and a bit from your help , we understood this solution will be inflexible so we decided to use the bag of words approach which we saw in class.

This way the dictionary will be build from all the samples we got, it would be more flexible and we could use an SVM algorithm that could build a 10 dimensional hyper plane which separates the celebrities.

we started by exploring the suggested models we found a very useful one - sklearn.

we than built a classifier which used the multinomial classification and got a good result of 79% success in classification.

we then deiced to check where is our biggest loss of classification and got -

politics errors: 0.390479360852 - errors between classifying politicians

others errors: 0.0 - errors between other categorize (not politicians)

between errors: 0.609520639148 - errors between those two groups

we started thinking about using first the classifier as separating two planes one of others (4,5,6,7,8,9)

and other of politicians (0,1,2,3,4) then maybe we could get ride off most of our errors.

so we did two rounds one of separating into two planes and

another separation we did is sending each group to a different classifier and separating it(inner separation between politicians and others) .

after this two training rounds we took the validation set and did the same for it with the hypothesis we got from the training part, unfortunately we saw that this only made the loss worse.

we gave it some thought and after looking in the net we found out the SGD classifier might make our loss smaller .after that change we got a loss of 83% success!!

we thought about quitting the university and offer ourselves to google.

but then we thought why not make our classifier even better - we mostly changed all the available properties of our classifier in order to see if it will make any difference (for example we found

that hober_square loss gives us better result) - we got to 85%

next we printed all the results in order to see what we have missed , we saw that even for us and not because of the hour it was very hard to classify which twitt was written by which celeb, but we did see that we have a lot of emojicons which some celebs write more frequently and are going to waste.

we looked up again and then started to focus on a new strategy with pre processing our data in a different way .we decided to separate words even punctuation that way we can classify better each celeb and his special twitt pattern. by this new strategy we succeed on separating the information better and also identify emojicons which where taken

in consideration this time:) we got to 87% :):)

we than looked again on our misclassified examples and saw that we really can't see the differentiate

we looked more on the internet and found the last improvement for our algorithm n_grams

which takes misspelled words and change them to the correct word after this step we got an 0.9 success rate, and on our way to google.

it is very important to us to mark that the code we submitted was really short and does not express all the struggles and long way we did until we got this result.