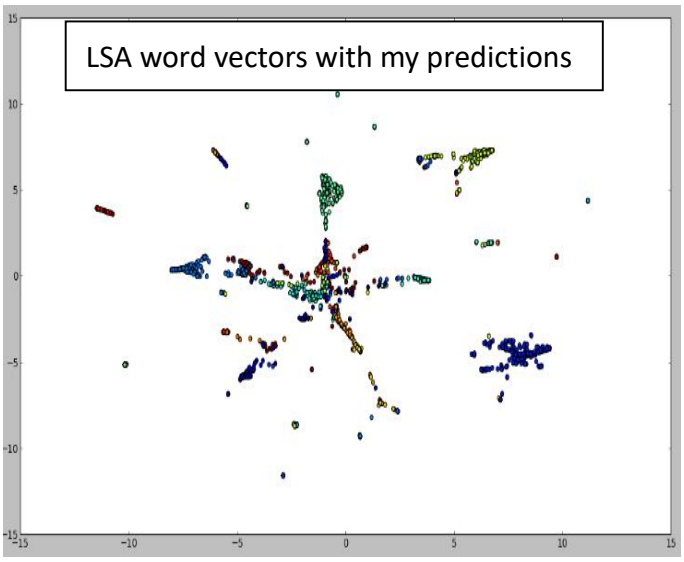
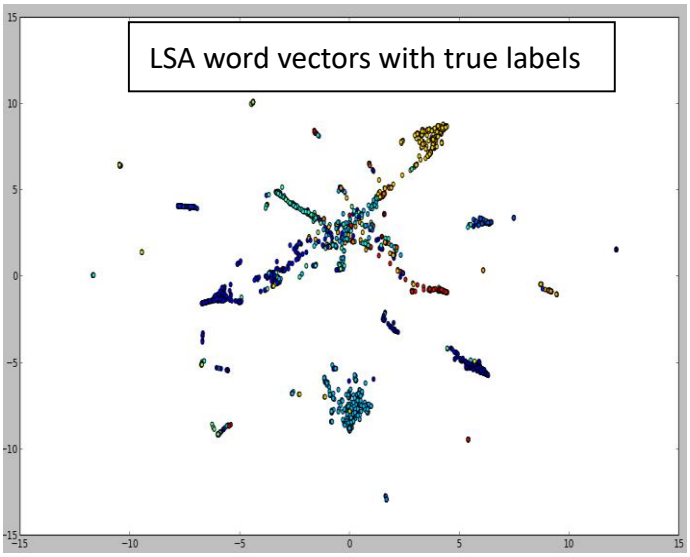
1. **Analyze the most common words in the clusters. Use TF-IDF to remove irrelevant words such as "the". (1%)**

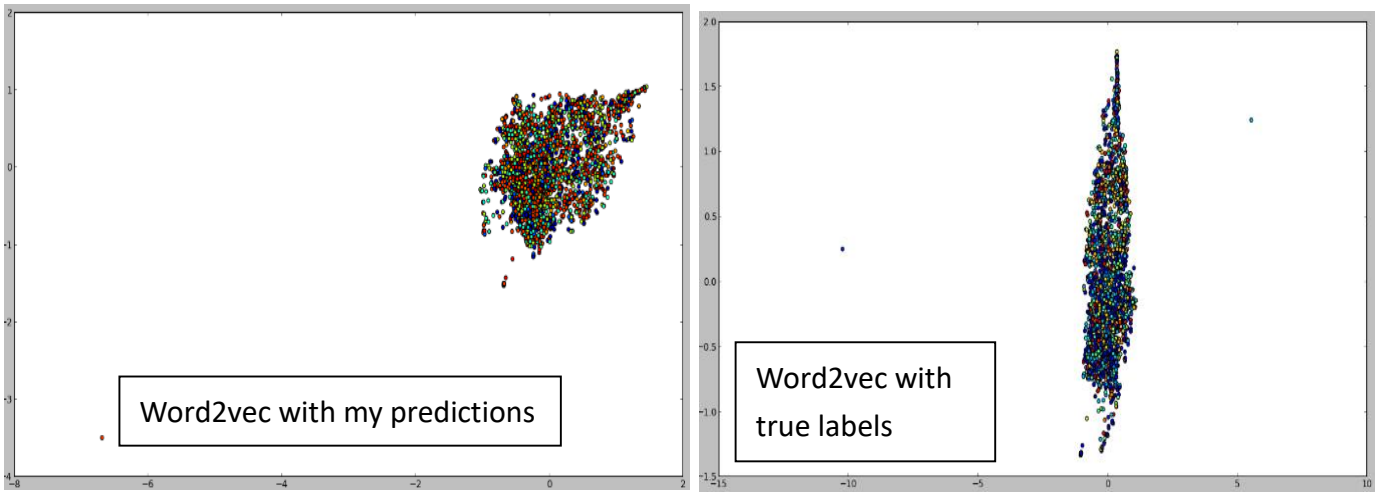
cluster 1: wordpress : 872
cluster 2: oracle : 763
cluster 3: svn : 606
cluster 4: apache : 609
cluster 5: excel : 863
cluster 6: matlab : 831
cluster 7: visual : 677
cluster 8: cocoa : 310
cluster 9: mac : 439
cluster 10: bash : 668
cluster 11: spring : 818
cluster 12: hibernate : 860
cluster 13: scala : 811
cluster 14: sharepoint : 742
cluster 15: ajax : 733
cluster 16: qt : 628
cluster 17: drupal : 862
cluster 18: linq : 859
cluster 19: haskell : 724
cluster 20: magento : 879

2. **Visualize the data by projecting onto 2-D space. Plot the results and color the data points using your cluster predictions. Comment on your plot. Now plot the results and color the data points using the true labels. Comment on this plot. (1%)**

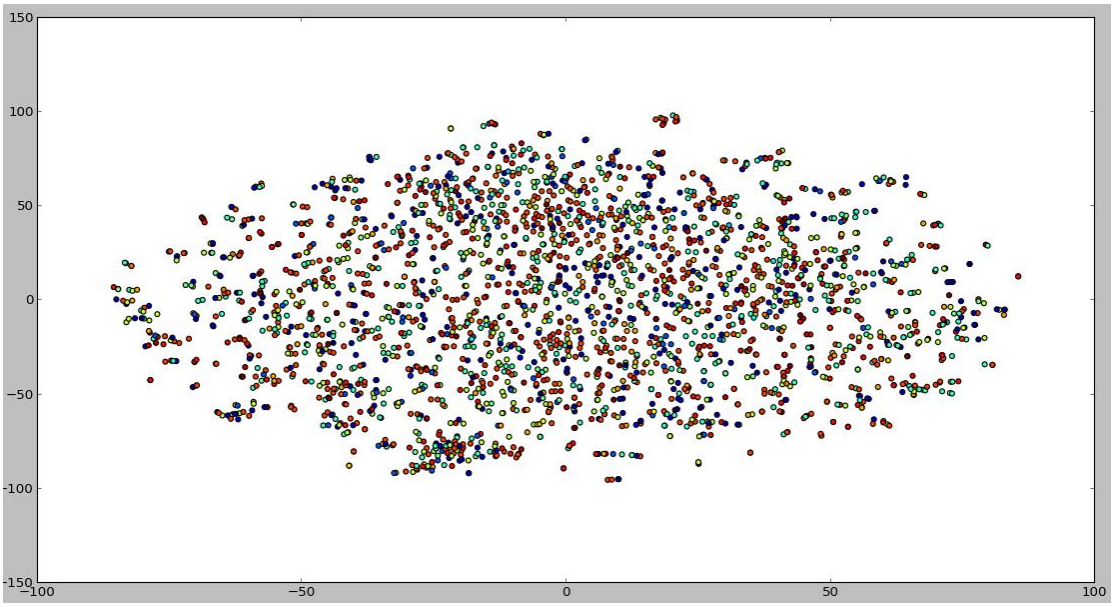


Here I use LSA with k = 35 to cluster “title_StackOverflow.txt”. I use t-SNE to plot high dimension vectors into 2D plots. Data points in the left graph is colored by my own prediction, while the ones in the right graph are colored by true labels. We can see some clear strides that would be clustered into different clusters. I suppose this means that most of the titles are related. Therefore, K-means clustering wouldn’t be the best choice for clustering.




These two figures presents the results of my word2vec approach. Clearly, it didn’t work. Later, I use 1000 iterations in t-SNE and get the following plot. We can see that titles weren’t clustered right. It is due to insufficiency of the pre-trained word2vec model, since we only use a relatively small document to train our word2vec model.



3. Compare different feature extraction methods. (2%)
   I tried the LSA approach and word2vec approach to extract features from titles. I used the sklearn LSA module with mini-batch K-means clustering. I also use

genism word2vec module for the word2vec approach and sklearn K-means clustering for the word2vec approach.
LSA did a better job at clustering data points with smaller vector space. Word2vec didn't succeed due to lack of training data for model building. It is surprising that LSA could do a better job in this case.

4. Try different cluster numbers and compare them. You can compare the scores and also visualize the data. (1%)
   Following is the figure of 20 clusters using LSA clustering comparing to 35 clusters. 35 clusters gives better accuracy in testing.