machine learning hw4 report

b02902080 資工四 郭傳駿

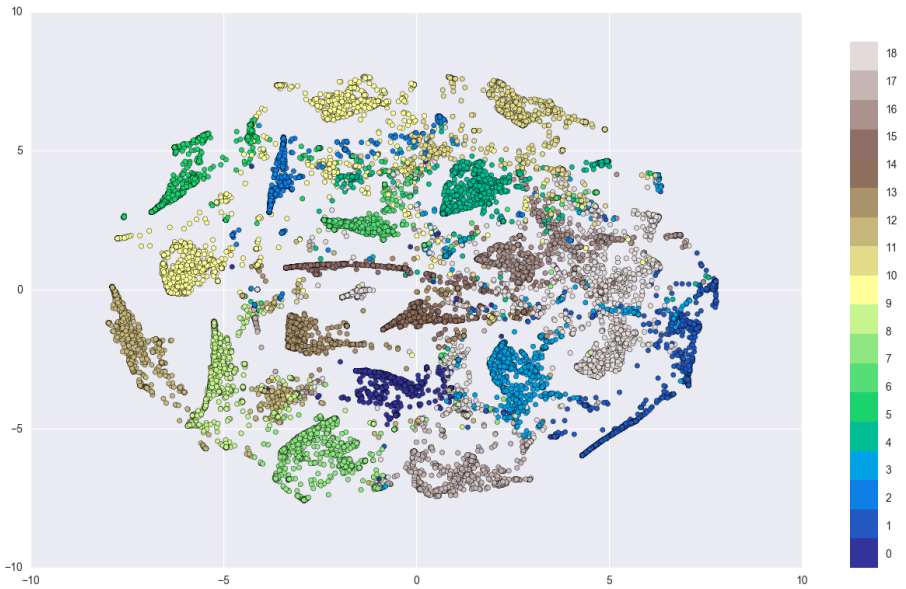
1. Remove stop-words & Analyze the most common words in each cluster

- The TfidfVectorizer in scikit-learn (sklearn.feature_extraction.text) takes a parameter “stop_words=‘english’ ” which removes stop words from each question title.
- The most frequent words in each cluster can be printed out by:
    1. calling get_feature_names() to get the mapping form index of each feature to feature themselves (the feature here refers to each unique english word)
    2. inverse_transform() the cluster_centers_ and sort the features in each clusters by frequency from high to low to get indexes of the most common features
    3. the most frequent words in each cluster can now be retrieved
- iteratively add additional words to the set of most common words in a cluster if the tf-idf frequency of the next common word is bigger than 0.9* of the previous one.
- 19 sets of common words:
  [['drupal'], ['excel'], ['sharepoint'], ['mac'], ['svn'], ['oracle'], ['hibernate'], ['ajax'], ['qt'], ['matlab'], ['bash'], ['linq'], ['spring'], ['apache'], ['wordpress'], ['haskell'], ['visual', 'studio'], ['scala'], [‘magento’]] I chose only 19 clusters when doing clustering because one set of the common words will be irrelevant (‘use’, ‘using’, ‘file’) if the clustering is done with 20 clusters.

2. visualize the data

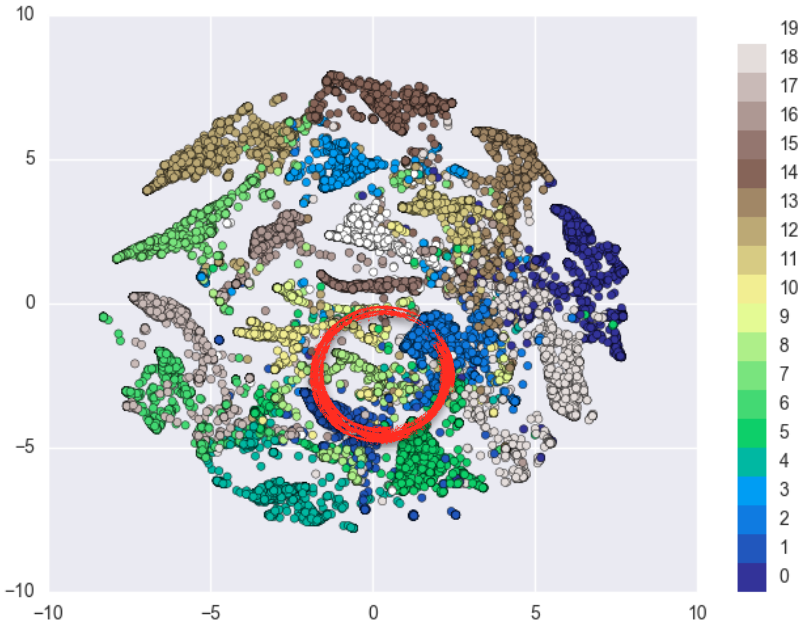
DATA VISUALISATION USING TSNE

- **PREDICTION WITH 19 CLUSTERS**



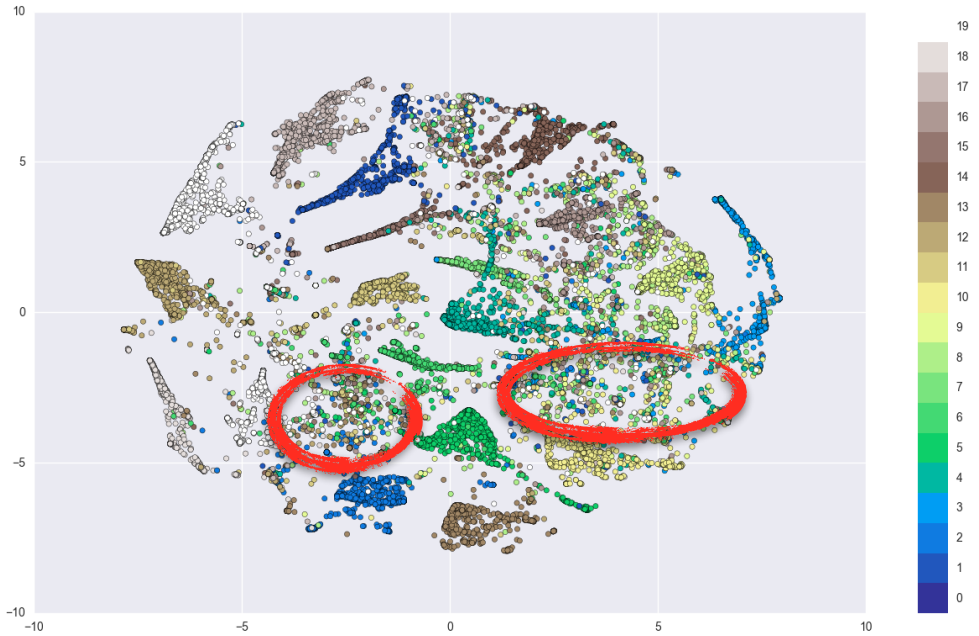
comment: data points in the same predicted class are bound tightly together, data points from different classes are separated nicely

- **PREDICTION WITH 20 CLUSTERS**



comment: data points are classified quite well, with slight ambiguity in area in the red circle

- **TRUE LABEL VISUALISATION**



comment: quite a lot of data points from different class close to each other as shown in the red circle, however, this may not be a full and true visualisation of the data since it is reduced to 2d.

3. compare between different feature extraction methods

- **TRAINED WITH 20 CLUSTERS**

PURE BAG OF WORDS
- performance: 0.90073
- top words per cluster:

[['cocoa', 'subversion', 'file'], ['haskell'], ['matlab'], ['linq'], ['mac'], ['sharepoint'], ['oracle', 'ajax'], ['hibernate'], ['magento'], ['qt'], ['bash'], ['using'], ['visual','studio'], ['apache'], ['scala'], ['drupal'], ['svn'], ['excel'], ['spring'], [‘wordpress']]

BAG OF WORDS WITH DIMENSION REDUCTION
- performance: 0.90970
- top words per cluster:

[['scala'], ['apache'], ['linq'], ['qt'],['magento'], ['wordpress'], ['drupal'],['matlab'], ['qt', 'apache', 'use', 'sharepoint'], ['using'], ['excel'], ['visual', 'studio'], ['svn'], ['bash'], ['haskell'],['sharepoint'], ['spring'], ['ajax'], ['hibernate'], [‘oracle’]]

PURE TF_IDF
- performance: 0.90451
- top words per cluster:

[['hibernate'], ['scala'], [‘wordpress'],['matlab'], ['excel'], ['ajax'], [‘bash'],['oracle'], ['mac'], ['linq'], ['drupal'] , [‘apache'], ['magento'], ['project'], ['spring'], ['haskell'], [‘cocoa'], ['sharepoint'], ['visual', 'studio'], ['svn']]

TF_IDF WITH DIMENSION REDUCTION
- performance:
- top words per cluster:

[['sharepoint'], ['magento'], ['use', 'scala'], ['haskell'], ['drupal'], ['spring'], ['ajax'], ['matlab'], ['apache'], ['oracle'], [‘linq'], [‘visual', 'studio'], ['bash'], ['mac'],['scala'], ['excel'], ['qt'], ['wordpress'], ['hibernate'], [‘svn']]

4. comparison between cluster numbers

In the previous section i demonstrated the top words of each of 20 clusters. Some are marked purple. These are words that are irrelevant to any subject(tag) and so shouldn't be counted as a cluster centroid. To address this problem, I used only 19 clusters instead. Here are the results.

- **TRAINED WITH 19 CLUSTERS**

PURE BAG OF WORDS

[['bash'], ['magento'], [‘excel'],['hibernate'], ['using'], ['svn'], ['oracle'], ['matlab'], ['haskell'], ['visual', 'studio'], ['drupal'], ['apache'], ['qt'], ['wordpress'], ['sharepoint'], ['linq'], ['scala'], ['ajax'], ['spring']]

BAG OF WORDS WITH DIMENSION REDUCTION

[['bash'], ['magento'], [‘excel'], ['hibernate'], ['using'], ['svn'], ['oracle'], ['matlab'], ['haskell'], ['visual', 'studio'], ['drupal'], ['apache'], ['qt'], ['wordpress'], ['sharepoint'], ['linq'], ['scala'], ['ajax'], [‘spring']]

— — — > it seems that irrelevant words still exists using Bag of Words method

PURE TF_IDF

[['qt'], ['oracle'], ['wordpress'], [‘scala'], ['drupal', 'sharepoint'], ['visual', 'studio'], ['excel'], ['magento'], ['svn'], [‘spring'], ['hibernate'], ['linq'], ['matlab'], [‘ajax'], ['content'], ['haskell'], ['apache'], ['file'], ['bash']]

TF_IDF WITH DIMENSION REDUCTION

[['scala'], ['excel'], ['ajax'], ['haskell'], ['drupal'], ['bash'], ['oracle'], ['matlab'], ['spring'], ['apache'], ['magento'], ['visual', 'studio'], ['wordpress'], ['linq'], ['qt'], ['sharepoint'], ['mac'], ['hibernate'], ['svn']]