COMP 527 − 2017 − 2 CA Assignment
Data Clustering
Implementing the k-means clustering algorithm

Assessment Information

Assignment Number	2 (of 2)
Weighting	13%
Assignment Circulated	3rd March 2017
Deadline	31st March 2017, 15:00 UK Time (UTC)
Submission Mode	Electronic via Departmental submission system
Learning outcome assessed	(1) A critical awareness of current problems and research issues in data mining.
Purpose of assessment Marking criteria	This assignment assess the understanding of k-means clustering algorithm by implementing k-means for text clustering. Marks for each question are indicated under the corresponding question.
Submission necessary in order to satisfy Module requirements?	No
Late Submission Penalty	Standard UoL Policy.

1 Objectives

This assignment requires you to implement the k-means clustering algorithm using the Python programming language.

> **Note that *no credit will be given for implementing any other types of clustering algorithms or using an existing library for clustering instead of implementing it by yourself. However, you are allowed to use numpy and scipy libraries for accessing data structures such as numpy.array or scipy.sparse. But it is not a requirement of the assignment to use numpy or scipy. You must provide a README file describing how to run your code to produce the results. Programs that do not run will result in a mark of zero!***

2 Text Clustering using k-means

In the assignment, you are required to cluster Amazon product reviews that belong to four product categories: *books*, *electronic appliances*, *dvds*, and *kitchen appliances*. Moreover, each category is further divided into positive-valued sentiment reviews and negative-valued sentiment reviews. In total, you will find reviews that belong to $4 \times 2 = 8$ categories in the data file provided for the assignment (Download *data.txt* from COMP 527 website).

The format of the data file is as follows. Each line of the data file corresponds to one review. The first element in the line represents the label of the instance (eg. *kitchen-positive* indicates that the review is a positive sentiment review about some kitchen appliance). The next elements (separated by spaces) in the line represent the unigram and bigram features extracted from the review. Note that the two words in a bigram feature are connected by two underscores. Reviews are represented using binary-valued features (i.e. each feature appears exactly once in a given line). You must **not** use the instance labels as features for the clustering algorithm. They must be used only for evaluation purposes.

Questions

(1) Write a program to load the data instances to memory from the provided file *data.txt*. (**10 marks**)

(2) Implement the k-means clustering algorithm with Euclidean distance to cluster the instances into k clusters. Make sure that you normalise each feature vector to unit L2 length before computing Euclidean distances. (**40 marks**)

(3) Vary the value of k from 2 to 20 and compute the macro-averaged precision, macro-averaged recall, and macro-averaged F-score for each set of clusters. (**20 marks**)

(4) Plot three graphs showing the value of k in the x-axis and each of the three evaluation measures (macro-averaged precision, macro-averaged recall, and macro-averaged F-score) in the y-axis in each graph. Briefly describe the trends that you can observe regarding the different evaluation measures and the value of k in your three graphs. (**12 marks**)

(5) Instead of selecting the mean in a cluster, select the instance that is closest to the mean as the cluster centre when performing k-means clustering. Compute the macro-averaged precision,

macro-averaged recall and macro-averaged F-score for k values from 2 to 20 for this modified version of the k-means algorithm and create three graphs separately with each evaluation measure. Briefly describe how this modification affects the performance of the k-means clustering algorithm. (**18 marks**)

3 Deadline and Submission Instructions

- Deadline for submitting this assignment is **31st March 2017, 15:00 UK time (UTC)**.
- Submit
  (a) the source code for all your programs,
  (b) a README file (plain text) describing how to compile/run your code to produce the various results required by the assignment, and
  (c) a PDF file providing the answers and graphs for the questions (4), and (5).

  Compress all of the above files into a single tar ball (tgz) file and specify the filename as *studentid.tgz*. It is extremely important that you provide all the files described above and not just the source code! (If you are unable to create a tgz file then create a zip file)
- Submission is via the departmental electronic submission system accessible (from within the department) from
  `http://intranet.csc.liv.ac.uk/cgi-bin/submit.pl?module=COMP527`.