Algorithm for evaluating the stability of clustering solutions

Representations:

* + F = { λi : i= 1,..., f }: The set of all features in a given dataset
  + f = |F|: The number of features in a dataset
  + k: The number of samples in the validation procedure
  + C: The number of clusters generated by the clustering algorithm
  + , m = 1,…, f: the probability (percentage) that a data point from the training dataset assigned to cluster i has the m\_th feature



* + , m = 1,…, f: the probability (percentage) that a data point from the testing dataset assigned to cluster i has the m\_th feature



1. Data Preparation:

1. Scrap data from the web (e.g GDELT)
2. Clean and preprocess the dataset.

2. Dataset Manipulation:

1. Randomly shuffle the original dataset.
2. Split the original dataset into k parts (folds).

3. Cross-validation Loop:

a. For each fold j=1,...,k:

1. Take fold j as the test dataset (each fold, in turn, is used as the test dataset).
2. ﻿ Take the remaining samples/folds together as the training dataset.
3. Generate keywords using KeyBERT/KeyLLM on I and II.
4. Refine and remove duplicated keywords/phrases.
5. Represent the generated keywords using LLM embedding.
6. Reduce the size of the embedding via UMAP/t-SNE.
7. Apply normalization to the embedding if needed.
8. Generate clusters on the reduced version of the training dataset.
9. Assign points from the test data to the corresponding clusters in step vii.

b. For each cluster i = 1, …, C:

* 1. Compute the empirical probabilities , m = 1,…, f of the occurrence of the features in cluster i based on the samples in the training dataset.



* 1. Compute the probabilities , m = 1,…, f of the occurrence of the features in cluster i using the assignment of the points from the test dataset to the clusters.



* 1. Compute the root mean squared error (RMSE) between the probabilities calculated in steps 1 and 2 (in the previous steps).

4. **Record the final scores:**

1. Once the loop in step 3 finishes:

* Note down the errors as a quality measure for cluster i obtained in fold j.
* Compute the average scores for each cluster across all the folds.
* Determine the global mean scores of all clusters across all samples.