Information Retrieval

Phase 5 Report

In this assignment, I performed document clustering on a corpus of HTML documents using the agglomerative clustering algorithm with the complete link method. The goal was to identify similarities and dissimilarities between documents and determine the closest document to the corpus centroid.

Implementation

Step-1: Tokenization and Preprocessing:

The input HTML documents were tokenized using the **simple_preprocess** function from the **gensim** library.

I removed HTML tags using regular expressions and applied additional preprocessing steps such as converting tokens to lowercase and removing stopwords.

The resulting tokens were used as the input for clustering.

Step-2: Similarity Matrix Calculation:

I calculated the similarity between documents using the cosine similarity metric.

A similarity matrix was constructed to store pairwise similarities between documents.

The similarity matrix was implemented as a 2D numpy array, where each entry represented the similarity between two documents.

Step-3: Agglomerative Clustering:

The agglomerative clustering algorithm with the complete link method was employed.

Initially, each document was treated as a singleton cluster.

At each step, the two clusters/documents with the highest similarity were merged.

The merging process was recorded to track the clusters being merged.

The clustering process continued until no two clusters/documents had a similarity greater than the specified threshold.

threshold = 0.4

Step-4: Cluster Naming Convention:

To assign names to the clusters, the merging process is tracked. Each cluster is represented by the indices of the documents it contains. When two clusters are merged, the new cluster retains the indices of both clusters. This allows us to keep track of the document indices associated with each cluster throughout the clustering process.

First 100 lines of Output:

```
PS C:\Users\srich\OneDrive\Desktop> py ir5.py
Merging clusters [0] and [12]
Merging clusters [0, 12] and [0, 12]
Merging clusters [1] and [44]
Merging clusters [1, 44] and [1, 44]
Merging clusters [2] and [7]
Merging clusters [2, 7] and [2, 7]
Merging clusters [3] and [13]
Merging clusters [3, 13] and [3, 13]
Merging clusters [4] and [32]
Merging clusters [4, 32] and [4, 32]
Merging clusters [5] and [6]
Merging clusters [5, 6] and [5, 6]
Merging clusters [8] and [31]
Merging clusters [8, 31] and [8, 31]
Merging clusters [9] and [22]
Merging clusters [9, 22] and [9, 22]
Merging clusters [10] and [34]
Merging clusters [10, 34] and [10, 34]
Merging clusters [11] and [89]
Merging clusters [11, 89] and [11, 89]
Merging clusters [14] and [82]
Merging clusters [14, 82] and [14, 82]
Merging clusters [15] and [99]
Merging clusters [15, 99] and [15, 99]
Merging clusters [16] and [69]
Merging clusters [16, 69] and [16, 69]
Merging clusters [17] and [110]
Merging clusters [17, 110] and [17, 110]
Merging clusters [18] and [101]
Merging clusters [18, 101] and [18, 101]
Merging clusters [19] and [24]
Merging clusters [19, 24] and [19, 24]
Merging clusters [20] and [109]
Merging clusters [20, 109] and [20, 109]
```

```
Merging clusters [20, 109] and [20, 109]
Merging clusters [21] and [129]
Merging clusters [21, 129] and [21, 129]
Merging clusters [23] and [61]
Merging clusters [23, 61] and [23, 61]
Merging clusters [25] and [68]
Merging clusters [25, 68] and [25, 68]
Merging clusters [26] and [152]
Merging clusters [26, 152] and [26, 152]
Merging clusters [27] and [55]
Merging clusters [27, 55] and [27, 55]
Merging clusters [28] and [185]
Merging clusters [28, 185] and [28, 185]
Merging clusters [29] and [76]
Merging clusters [29, 76] and [29, 76]
Merging clusters [30] and [45]
Merging clusters [30, 45] and [30, 45]
Merging clusters [33] and [71]
Merging clusters [33, 71] and [33, 71]
Merging clusters [35] and [50]
Merging clusters [35, 50] and [35, 50]
Merging clusters [36] and [157]
Merging clusters [36, 157] and [36, 157]
Merging clusters [37] and [147]
Merging clusters [37, 147] and [37, 147]
Merging clusters [38] and [106]
Merging clusters [38, 106] and [38, 106]
Merging clusters [39] and [170]
Merging clusters [39, 170] and [39, 170]
Merging clusters [40] and [230]
Merging clusters [40, 230] and [40, 230]
Merging clusters [41] and [164]
Merging clusters [41, 164] and [41, 164]
Merging clusters [43] and [199]
Merging clusters [43, 199] and [43, 199]
Merging clusters [46] and [175]
Merging clusters [46, 175] and [46, 175]
Merging clusters [47] and [87]
Merging clusters [47, 87] and [47, 87]
Merging clusters [48] and [102]
Merging clusters [48, 102] and [48, 102]
```

```
Merging clusters [39, 170] and [39, 170]
Merging clusters [40] and [230]
Merging clusters [40, 230] and [40, 230]
Merging clusters [41] and [164]
Merging clusters [41, 164] and [41, 164]
Merging clusters [43] and [199]
Merging clusters [43, 199] and [43, 199]
Merging clusters [46] and [175]
Merging clusters [46, 175] and [46, 175]
Merging clusters [47] and [87]
Merging clusters [47, 87] and [47, 87]
Merging clusters [48] and [102]
Merging clusters [48, 102] and [48, 102]
Merging clusters [49] and [154]
Merging clusters [49, 154] and [49, 154]
Merging clusters [51] and [373]
Merging clusters [51, 373] and [51, 373]
Merging clusters [52] and [95]
Merging clusters [52, 95] and [52, 95]
Merging clusters [53] and [215]
Merging clusters [53, 215] and [53, 215]
Merging clusters [54] and [67]
Merging clusters [54, 67] and [54, 67]
Merging clusters [56] and [91]
Merging clusters [56, 91] and [56, 91]
Merging clusters [57] and [131]
Merging clusters [57, 131] and [57, 131]
Merging clusters [58] and [103]
Merging clusters [58, 103] and [58, 103]
Merging clusters [59] and [189]
Merging clusters [59, 189] and [59, 189]
Merging clusters [60] and [64]
Merging clusters [60, 64] and [60, 64]
Merging clusters [62] and [126]
Merging clusters [62, 126] and [62, 126]
Merging clusters [63] and [150]
Merging clusters [63, 150] and [63, 150]
Merging clusters [65] and [437]
Merging clusters [65, 437] and [65, 437]
Merging clusters [66] and [148]
Merging clusters [66, 148] and [66, 148]
```

Method for Giving Names to Clusters: The names for the clusters are assigned based on the index of the representative document within each cluster. The representative document is assumed to be the first document in the cluster. The cluster names are derived by adding 1 to the index since indexing starts from 0. Therefore, the cluster names correspond to the document numbers in the corpus.

Implementation of the Similarity Matrix and Major Data Structures:

Similarity Matrix: The similarity matrix is implemented as a two-dimensional NumPy array named similarity_matrix. It is initialized with zeros and has dimensions (num_documents, num_documents) to store the pairwise similarity values between documents. The similarity matrix is calculated in the calculate_similarity_matrix function by iterating over the tokens of each document pair and calling the calculate_cosine_similarity function.

Data Structures:

tokens: It is a list of lists where each inner list represents the tokenized text of a document. It is generated by the generate_tokens function by reading the HTML files from the input directory, performing tokenization, and applying preprocessing steps.

clusters: It is a list of lists where each inner list represents a cluster. Initially, each document is considered as a separate cluster. The perform_clustering function merges clusters iteratively until the similarity falls below the threshold. The clusters list is updated during the merging process.

merging_process: It is a list that keeps track of the merging steps during clustering. Each entry in the list represents the merging of two clusters/documents. The merging process is recorded by appending tuples of cluster indices to the merging_process list.

doc1_vector and doc2_vector: These are defaultdicts from the collections module used to store the term frequencies of tokens in two documents. They are created and populated in the calculate_cosine_similarity function.

Which pair of HTML documents is the most similar?

The pair of HTML documents that are the most similar can be determined by examining the merging process during clustering. The first pair of documents that are merged into a cluster indicate the highest similarity. By printing the merging process, we can identify the first merged clusters and determine the pair of documents that are most similar.

In this case the most similar pair of HTML documents are ([0],[12])

Which pair of documents is the most dissimilar?

The pair of documents that is the most dissimilar can be identified by examining the last entry in the merging process list. The last entry represents the final merging step where all clusters/documents have been merged. Therefore, the last entry in the merging process list represents the pair of documents that is the most dissimilar.

In this case the most dissimilar pair of documents are ([285, 296], [285, 296])

Which document is the closest to the corpus centroid?

To determine the document closest to the corpus centroid, we need to compute the similarity between each document and the centroid. The centroid can be calculated as the average of the similarity matrix's rows or columns. We can find the document closest to the centroid by finding the row or column in the similarity matrix with the highest similarity value to the centroid.

In this case the document closest to the corpus centroid is 1.

Output:

```
Merging clusters [295] and [331]
Merging clusters [295, 331] and [295, 331]
Merging clusters [285] and [296]
Merging clusters [285, 296] and [285, 296]
Most Similar Pair: ([0], [12])
Most Dissimilar Pair: ([285, 296], [285, 296])
Closet document to corpus centroid: 1
```

Result:

In this analysis, I have performed document clustering on a corpus of HTML documents. The code successfully generates a similarity matrix, performs hierarchical agglomerative clustering using the complete link method, and provides information about the most similar, most dissimilar, and closest document to the corpus centroid. The implementation relies on tokenization, cosine similarity calculation, and tracking the merging process to assign names to clusters.