Document Clustering

The Challenge

Using the dataset contained in data.zip, extract the text from each document and determine:

1. The best number of clusters, N, for the dataset
2. Which documents are assigned to each cluster

You can verify that the documents you have received are correct by matching the md5 checksum of the zip folder: 0be8e15b0a64ad4b87c9cc50ff4e1b60

The final result should be output to a file called results.json with the below structure:

{

0: [file1, file2, file3, ..., fileN],

1: [file1, file2, file3, ..., fileN],

...

N: [file1, file2, file3, ..., fileN]

}

where each key is a cluster numbered from 0-N and the value is a list of filenames assigned to that cluster. Do not include the file path or extension in the file name. The cluster order is arbitrary, what matters are the documents contained within.

Rules

1. Python is preferred for this coding challenge, but if you feel unable to complete it using Python, use whatever programming language you feel most comfortable in.
2. You may use any packages that exist but you must be able to explain what they are doing.

Going beyond

Bonus Points For:

1. Using only well known and reputably sourced packages.
2. Using more than one method to determine the best number of clusters.
3. Using (and comparing) more than one clustering algorithm.
4. Well documented code.

Last Details

**If you have any questions, please do not hesitate to reach out via email.** Use git to track your progress, and push your solution to a github repository (public or if private, we will need access). This project should take around four hours, but we will respect your time. Please take as much or little time as needed, you will only be evaluated on correctness, not time.