1. Knowing that HDBSCAN has a complexity of n2, what are the big-data-related bottlenecks of the above analysis?
   * Describe in detail how you'd handle each bottleneck.
2. What improvements would you make to the analysis above? From an analytical perspective rather than a computational one.
3. How would you handle instances assigned a -1 by HDBSCAN? I.e., 1. would you throw them away or would you assingn them back into the cluster by using "some approach or heuristitic" to do so? If you chose 1, then describe why? If you chose 2., describe how?

Answer: I will take a heuristic approach to assigning them back into a cluster. I can clean the data by lowering the cases and removing stopwords, punctuations, and non-utf characters. Important words will remain after data cleaning. This will help the algorithm to calculate the data based on important information, so the result is expected to be more accurate. Another approach is to add dimensions so that more features of the data will be captured and serve as important information for the algorithm while it is operating.

1. K-means would work very well with the cosine distance. How would you go about findind the best value of k?

Answer: I can use the elbow method to find the best value of K. As the number of K is increased, more variances will be explained by the algorithm. The sume of squared of error(SSE) will be reduced by increasing the number of K. In a line chart, the X-axis is the number of K and the Y-axis is the SSE. The line will be dropping while the number of K is increased, but it will stop dropping and become flat when the K is increased to a certain point. That is the turning point that is considered as the best value of K because adding more K data points will not significantly explain more variances.

1. Discribe the approch you'd use to reduce the number of topics obtained?

Answer: One possible solution is to pick some data points from two clusters and compute the cosine similarity for instance between the data points of the two clusters. If the cosine similarity is high, it means the data points of the two clusters are similar. Thus, we are confident to merge the two clusters together.

1. Satisfied with your analysis (sentiment analysis), you decide to scale it for your company to run on massive amounts of data daily. To do so, you decide to use the Amazon cloud. Your job here is to explore the various web services offered by amazon, identify those that are most relevant to each step in the analysis and descrive how and why you chose to include it in your pipeline. Your company decide that uploading the data to Pinecone daily is costly (Amazon charges by GB of data transferred), so you will also need to identify a service that can do something similar to what you did with Pinecone. Note that Amazon does not have a vector (or embedding DB), so you need to find a service within the Amazon ecosystem that provides nearest neighbor search.

Provide an overview diagram that describes the services you will be using and how the data will flow between each service. For each service in your diagram, describe which task this service is doing and why you chose it. You can find a list of AWS procts [here.](https://aws.amazon.com/products)

Cloud: data to be analyzed are stored.

Download the data from a cloud to Amazon S3

**Amazon S3**: It is an object storage service. Any amount of data can be stored and protected in this cost-effective storage classes. Therefore, it is chosen for the large data processing.

Use Amazon Textract to extract data from Amazon S3.

**Amazon Textract**: It is a machine learning service to automatically extract text.

Use Amazon OpenSearch Service to perform the same function as Sentence Transformer to produce embeddings.

**Amazon OpenSearch Service**: It provides fast access and response to large amounts of data. It also provides k-nearest neighbors(knn) search.

Use Amazon OpenSearch Service to perform the same function as Sentence Transformer to produce embeddings.

**Amazon Comprehend**: It can process unstructured text to extract sentiment. Once a query is sent to the algorithm, a sentiment score of mixed, positive, negative, and neutral will be displayed for that query, and the highest score will be chosen to represent the sentiment of the document.

Store the results in data warehouse for further retrieval. Run queries using the query editor on the Amazon Redshift console.

**Amazon Redshift**: It is a data warehouse to analyze structured and semi-structured data across data warehouses and operational databases. It can deliver performance at any scale. For example, the service can support unlimited concurrent users and queries.