Creating Semantic Spaces Using Document Clustering

By: Tristan Weger, Ryan Rubadue, Yahya Emara

Project Goals

Goals:

- Create a representation of natural language that captures meaning (semantic space) using long documents.
- Compare different embedding methods (sentence, segment).
- Display the semantic space of large dimensionality with a simplified 2 dimensional map.
- Create a cognitive map that is easy to interpret and understand.

Intellectual Merits

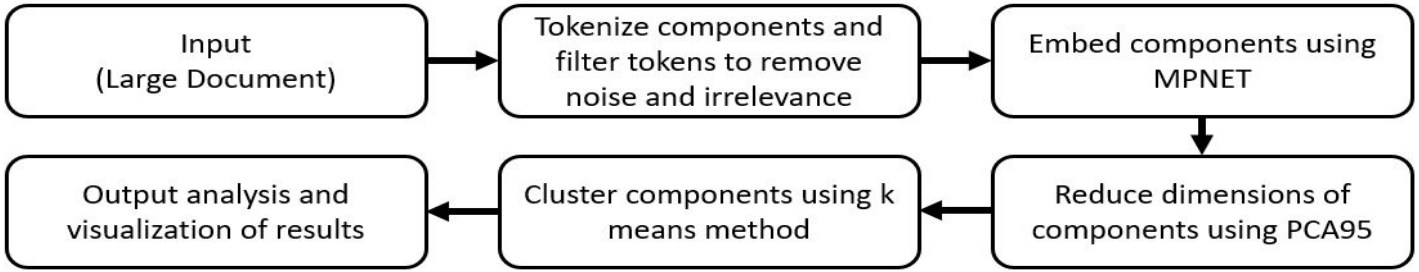
- Comparison of segment and sentence tokenization methods
  - Advantages and disadvantages of these methods are determined
- Large documents are composed of many topics
  - the optimal number of clusters is calculated using the silhouette method metric
- Information extraction from large documents contains lots of noise
  - This noise is properly parsed and filtered to get the most accurate results
- This system uses a combination of machine learning and NLP methods including umap, nltk, and MPNET

Broader Impacts

- There are many practical applications for this system.
  - Brainstorming - thinking of new ideas is challenging and time consuming, this system can take in ideas and output new similar ideas, preserving the problem solving process and saving time.
  - Text or article summarization - this system creates component embeddings and clusters that can be applied to a summarization system to create comprehensible shortened summaries of large documents or bodies of text
  - Language generation - this system creates a semantic space which is a representation of human language that creates meaning. This meaning can be applied to create a system that can generate accurate and coherent language.

Design Specifications

- A dataset of information must first be selected and input into the system.
- Basic text filtering including short sentence removal is performed.
- Sentences are tokenized and embedded into vectors of 768 dimensions using MPNET.
- Further filtering is done to remove irrelevant sentences.
- PCA 95 reduces the component dimensions to ~220.
- K means clustering is performed and mapped onto a 2D plot using UMAP.
- Word clouds are produced from clusters.



Technologies

- Technologies Developed can be split into:
- Text Filtering and Parsing Algorithms
  - Implementing **NLTK** techniques for Sentence Tokenization
- Embedding and Segmentation Algorithms
  - Using the **sentence transformers library** and the **MPnet-base-v1** model to embed the sentences into 768 dimensions
  - Separating the sentences into segments of related sentences using a segmentation algorithm that develops an accumulated score matrix and backtracking to determine the optimal segment splits while traversing the matrix
- Dimensionality Reduction Algorithms
  - **PCA-95** and **PCA-90** algorithms to reduce dimensionality from 768 to 200-220 dimensions
- Clustering Algorithms
  - Developed a **K-means** clustering function to cluster sentences onto a 2D semantic space
  - To generate a visual outcome, we used **Umap** algorithm to reduce dimensionality to 2D
  - Additionally, developed a word-cloud generating function to produce the most important words from each cluster, differing in size depending on how important a word is

Milestones

- Broadview of deliverables for the project:
  - Finding a dataset that fits requirements of long documents that discuss various topics
  - Parsing and filtering the documents into clean sentences
  - Finding a suitable embedding algorithm and implementing it
  - Implementing a segmentation algorithm for the embeddings and testing for accuracy
  - Producing cosine-similarity matrices and heatmaps to compare segments
  - Normalization or standardization of results to improve performance
  - Filtering irrelevant sentences to improve performance
  - Implementing PCA-95 to reduce dimensionality
  - Using K-means clustering to obtain meaningful clusters from the data
  - Implementing Umap algorithm to produce a 2D visual outcome
  - Producing meaningful word-clouds for each cluster

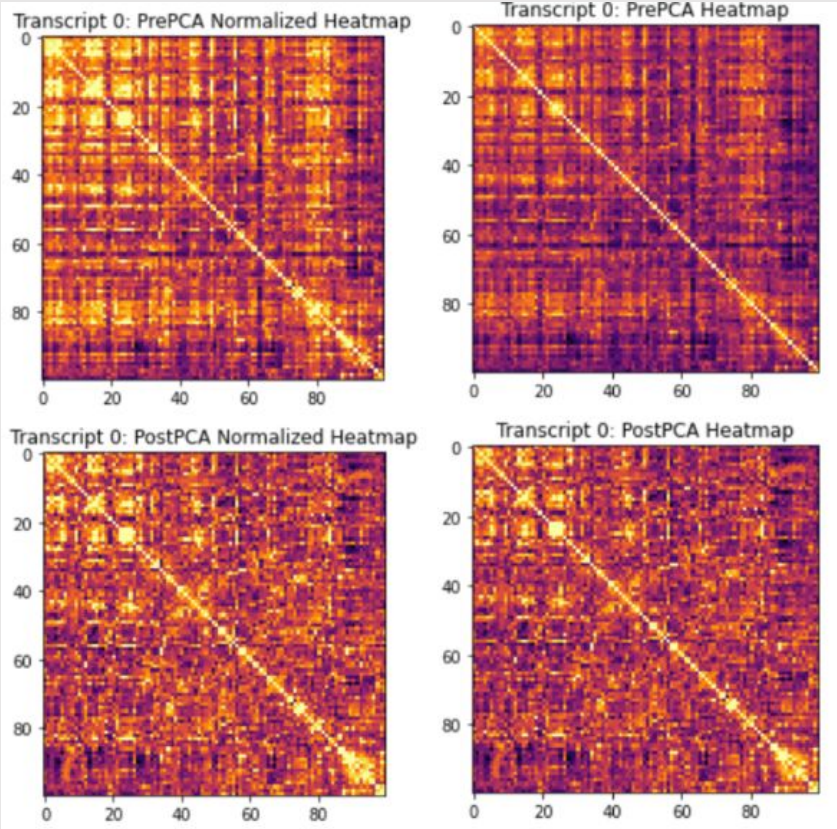
Results

Currently:

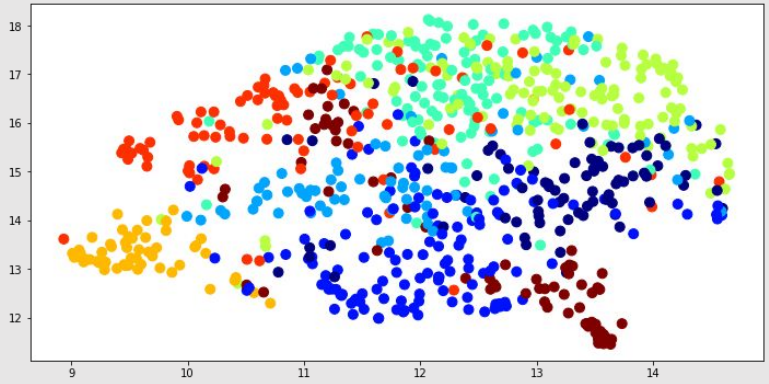
- Implementation of semantic analysis process when analyzing the text corpus at sentence level
- Visualizations/metrics for each portion of semantic analysis process
- Base level word clouds visualizations

By Expo:

- Comparison between relative performance of process at each level
- Improved clustering/visualizations
- Analysis/optimization of sentence filtering methodologies



Heatmaps for George Washington's Speeches



Sentence clusters of George Washington's Speeches (8 Clusters)

Examples of Word Clouds Created for George Washington

Challenges

- Managing extremely long application runtime
  - Google Colab used for executing application as a result
  - Review of application bottlenecks/potential time optimizations
  - Working with longer documents necessary for project goals
- Exposure to several natural language processing concepts
  - Dimension reduction methodologies, advanced clustering methods beyond k means
- Determining ‘useless’ tokens within document
  - Simple at word level, at sentence level much more difficult due to ambiguity