Exploring the Embedding Space for Enhanced RAG System Performance

A Statistical Approach to Understanding and Improving Retrieval-Augmented Generation

Gaurav Pooniwala

December 3, 2023

Introduction

- **Objective:** To explore and understand the embedding space of a RAG system to enhance its performance using statistical methods.
- **Research Question:** How can visualizing and analyzing the embedding space help us understand the RAG algorithm and use this understanding to improve its performance?

Context

- **Retrieval-Augmented Generation (RAG):** A method that combines the retrieval of relevant documents with the generation of responses using a language model.
- **Mechanism of RAG:** RAG identifies the nearest neighbor to a question or query from a database of documents stored as embeddings.
  - **Embeddings:** Dense vector representations of words or sentences. In this analysis, we use embeddings of size 1536.
  - **Role of Cosine Distance:** The similarity between the query and the documents is calculated using cosine distance, helping to identify the most relevant document.

Motivation

- **Importance of Analysis:** Understanding the embedding space and the effectiveness of cosine distance in finding relevant documents is crucial for improving RAG performance.
- **Understanding the Embedding Space:** Visualizing embeddings helps us gain insights into how words and sentences are represented in the latent space.
- **Improving RAG Performance:** Analyzing the embedding space can help identify ways to enhance the performance of RAG systems.
- **Analysis Goals:** Our goal is to understand the structure of the embedding space and evaluate the effectiveness of different clustering methods.

Data Description

- **Data Source:** Synthetically generated data based on typical datasets. Internal datasets were also used but results are not shared due to GDPR and proprietary constraints.
- **Data Types:**
  - **Word Embeddings:** Common words, nouns, verbs, and adjectives.
  - **Sentence Embeddings:** Question-answer pairs.
- **Data Size:** Analysis was run repeatedly in groups of up to 40 words and 16-20 question-answer pairs, totaling about 100 questions.

Methods Overview

- **Word Embeddings:**
  - Generate embeddings for common words, nouns, verbs, and adjectives.
  - Visualize the embeddings using t-SNE and PCA to understand the structure of the embedding space.
  - Test clustering methods to identify patterns in the embeddings.
- **Sentence Embeddings:**
  - Use a set of question-answer pairs to generate sentence embeddings.
  - Visualize the embeddings and analyze cosine distances to evaluate the accuracy of identifying relevant answers.

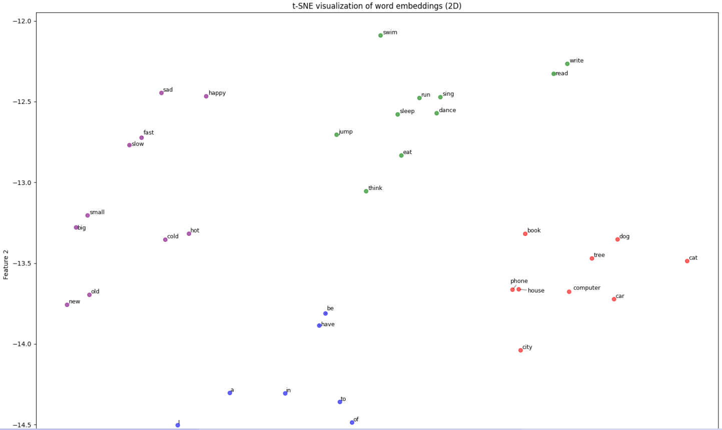
Word Embedding Analysis

We generate the embeddings for the following set of words using the text-embedding-ada-002 model:

- **Common Words:** the, be, to, of, and, a, in, that, have, I
- **Nouns:** cat, dog, house, car, tree, book, phone, computer, city, ocean
- **Verbs:** run, jump, eat, sleep, write, read, swim, dance, sing, think
- **Adjectives:** happy, sad, big, small, fast, slow, hot, cold, new, old

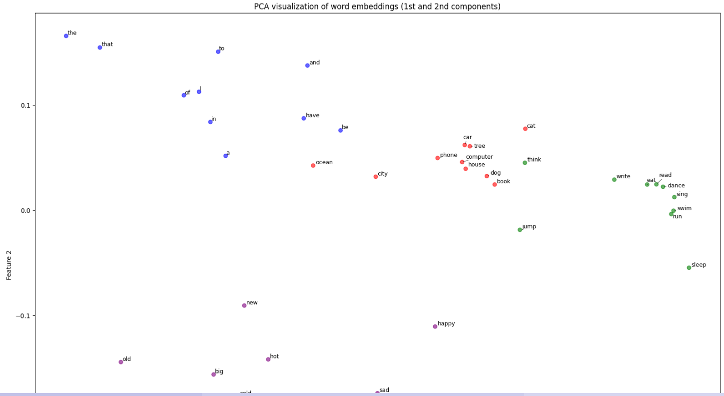
Word Embedding Analysis - t-SNE

- **Process:**
  - Perform t-SNE to reduce dimensions to 2D and 3D.
  - Visualize the embeddings.
- **Results:** Word embeddings of similar types are generally clustered together in the t-SNE visualization.



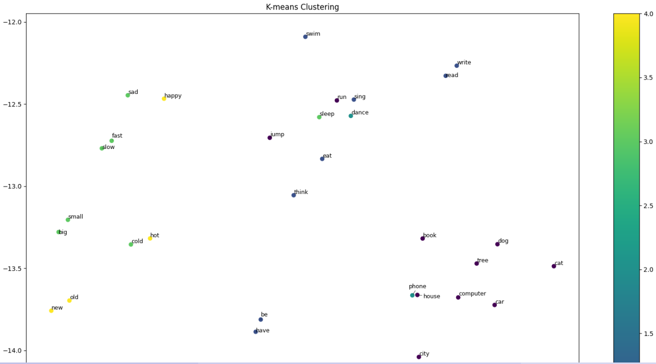
Word Embedding Analysis - PCA

- **Process:**
  - Perform PCA on the embeddings.
  - Visualize the first four principal components.
- **Results:** Word embeddings of similar types are generally clustered together in the PCA visualization. Although, the first two components explain only 13% of the variance.



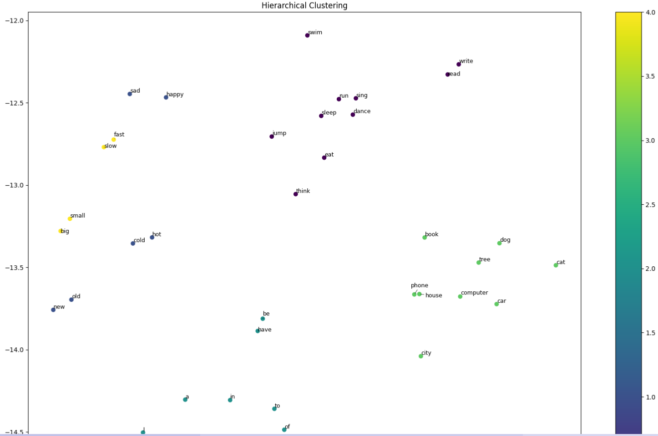
Word Embedding Analysis - Clustering (K-means)

- **Process:**
  - Perform K-means clustering.
  - Visualize the clusters.
- **Results:** K-means clustering results.
- K-means assumes clusters are spherical and equally sized, which may not be suitable when using cosine distances instead of euclidean distances.



Word Embedding Analysis - Clustering (Hierarchical)

- **Process:**
  - Perform hierarchical clustering.
  - Visualize the clusters.
- **Results:** Hierarchical clustering seems to perform marginally better than K-means.



Sentence Embedding Analysis

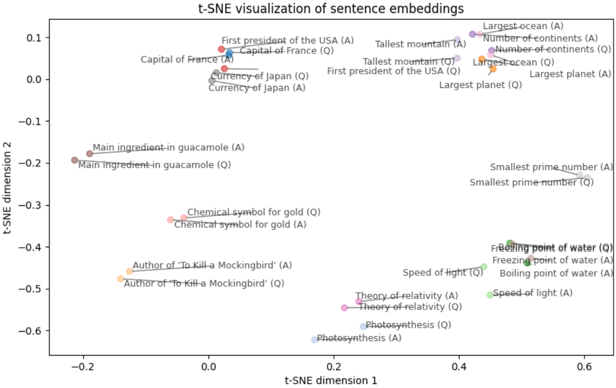
- **Objective:** To evaluate the effectiveness of sentence embeddings in identifying relevant answers using cosine similarity and clustering methods.
- **Process:**
  - Generate embeddings for a set of question-answer pairs.
  - Calculate pairwise cosine similarities to evaluate the relevance of answers.
  - Apply t-SNE for dimensionality reduction and visualize the embeddings.
  - Perform K-means and hierarchical clustering to group similar sentences.

Example Similarity Analysis

- **Relevant Similarity:**
  - Sentence 1: What is the capital of France?
  - Sentence 2: The capital of France is Paris.
  - Similarity: 0.74
- **Random Similarity:**
  - Sentence 1: What is the capital of France?
  - Sentence 2: Photosynthesis is the process by which green plants use sunlight to synthesize foods from carbon dioxide and water.
  - Similarity: -0.12
- **Accuracy:** The accuracy of finding the closest answer based on cosine similarity is 100% for our small dataset of 16 questions.

Sentence Embedding Analysis - t-SNE

- **Process:**
  - Perform t-SNE to reduce dimensions to 2D.
  - Visualize the embeddings.
- **Results:** Show plot of t-SNE visualization.



Sentence Embedding Analysis - Clustering (K-means)

- **Process:**
  - Perform K-means clustering on sentence embeddings.
  - Visualize the clusters.
- **Results:**
  - K-means clustering accuracy: 94%

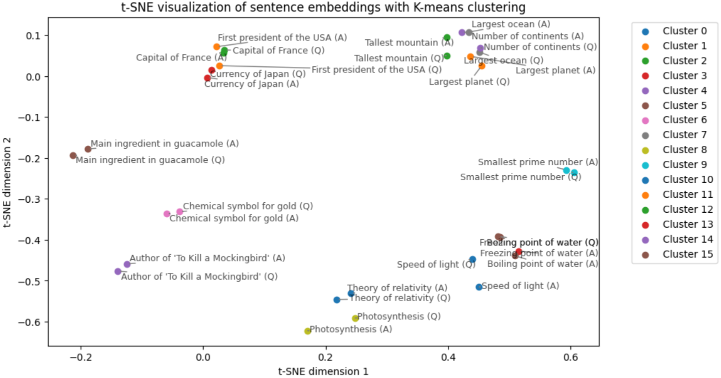


Figure: K-means Clustering of Sentence Embeddings

Sentence Embedding Analysis - Clustering (Hierarchical)

- **Process:**
  - Perform hierarchical clustering on sentence embeddings.
  - Visualize the clusters.
- **Results:**
  - Hierarchical clustering accuracy: 88%

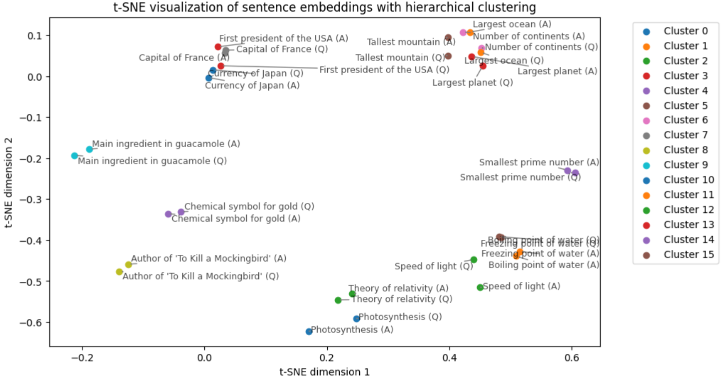


Figure: Hierarchical Clustering of Sentence Embeddings

Evaluation of Cosine Similarity and Clustering Methods

- **Process:**
  - Calculate cosine similarity for each question-answer pair.
  - Identify the closest answer based on cosine similarity.
  - Compare with K-means and hierarchical clustering on sentence embeddings.
- **Results:**
  - Cosine similarity accuracy: 100%
  - K-means clustering accuracy: 94%
  - Hierarchical clustering accuracy: 88%

Observations

- **Embedding Space:**
  - The embedding space is non-linear. Cosine similarity is used to calculate distances between embeddings.
  - The embedding space prioritizes grouping words and sentences of similar types together, such as verbs, nouns, or topics like geography and physics.
  - Opposite sentiments, emotions, or even opposite words tend to be very close to each other in the embedding space.
- **Methods:**
  - PCA, being a linear technique and the first 2 dimensions only accounting for 13% of the variance, is not ideal but still shows promising results.
  - t-SNE works well to visualize the data as a whole on the macro level, but individual distances are too warped to be interpreted directly.
  - The assumptions for both K-means and hierarchical clustering do not work very well for cosine distances. While K-means failed at the word level, hierarchical clustering partially failed at the sentence level.

Limitations

- **Synthetic Data:** Easy to identify the best question-answer pair. Real-world data may not have direct answers or may have multiple candidates.
- **Ground Truth:** May be unknown, requiring human validation.
- **Qualitative Testing:** Current testing for larger datasets is qualitative. Quantitative analysis is needed for real-world applications.
- **Scalability:** High accuracy achieved with simple question-answer pairs may not extend well to larger datasets:
    - More candidate answers reduce the probability of finding the correct answer.
    - Multiple correct or relevant answers may exist.
    - Qualitative analysis is not feasible for large datasets.

Potential Next Steps

- **Extensions:**
  - Use an LLM to automatically analyze the relevance of selected answers.
  - Test various strategies such as different embeddings and chunk sizes.
- **Improvements:**
  - Develop methods for quantitative analysis to enhance the current qualitative approach.

Conclusion

- **Summary:**
  - We explored the embedding space of a RAG system using statistical methods.
  - Visualizing and analyzing the embedding space helped us understand the RAG algorithm and identify ways to improve its performance.
- **Final Thoughts:**
  - The embedding space is non-linear, making classical techniques less effective.
  - Future work should focus on quantitative analysis and real-world data applications.

Questions

Q&A: Thank you for your attention. I am happy to answer any questions you may have.