

# APS 2: Vector-Based Search System Using BERT and Autoencoder

## Step 1: Find Embeddings

### Dataset Description

The dataset consists of a collection of essays by Paul Graham, the same dataset utilized in APS1. This dataset is composed of individual text documents covering a variety of topics, predominantly related to technology, entrepreneurship, and programming. Paul Graham's essays, along with the influence of Y Combinator, have shaped the perspectives of tech founders and startup culture, providing foundational insights for entrepreneurs worldwide.

### Embedding Generation Process

To capture the semantic structure of Paul Graham's essays, a pre-trained BERT model was used to generate initial embeddings. Text preprocessing involved lemmatization and removing non-informative stopwords to clarify meaning. Each document was tokenized and represented as a weighted average of token embeddings, with an attention mask emphasizing relevant tokens. This initial 768-dimensional embedding was then fine-tuned for the dataset using an autoencoder with intermediate layers to gradually reduce the dimensionality to a compact 128 dimensions. The autoencoder architecture employed GELU activation for smoother gradient flow, LayerNorm for stable training, and Dropout to minimize overfitting. The training was configured with key hyperparameters: a learning rate of 0.0001, weight decay of 1e-5, and a batch size of 16. The training schedule applied Cosine Annealing with Warm Restarts, with early stopping to prevent overfitting, resulting in embeddings that retain both syntactic structure and semantic detail.

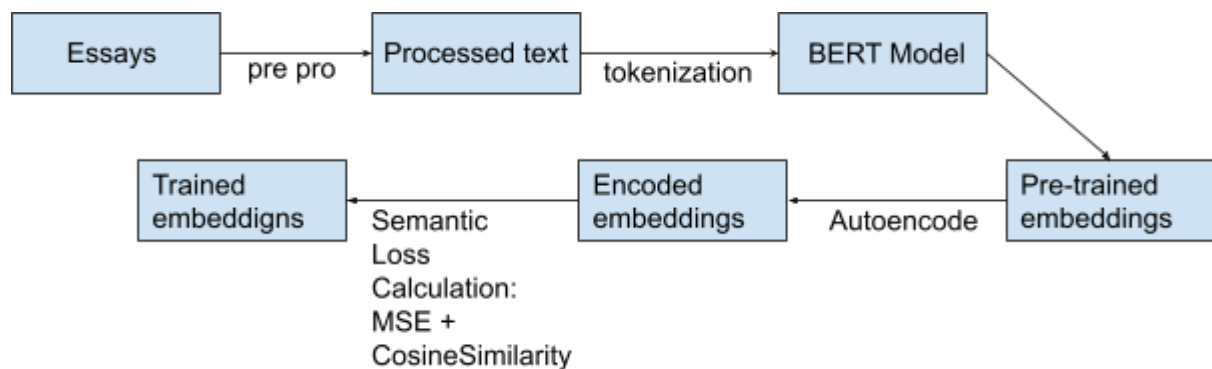


Figure 1: Neural Network Topology

## Training Process

The training process was structured to manage the high semantic overlap in Paul Graham's essays, where recurring themes could lead to overfitting. Embeddings were first generated using BERT and processed through a semantic autoencoder with a hybrid loss function combining Mean Squared Error (MSE) and Cosine Similarity. The MSE component minimized reconstruction error by reducing the difference between original and reconstructed embeddings, while Cosine Similarity preserved semantic alignment in the vector space, essential for maintaining meaningful relationships across thematically similar essays. The autoencoder's architecture, designed to gradually compress dimensionality from 768 to 128, included GELU activations to enhance gradient flow, LayerNorm to stabilize each layer, and Dropout to prevent overfitting on recurrent topics. This combined approach was meant to preserve both syntactic structure and essential semantic details, optimizing the embeddings for reliable semantic search across the essays.

### Loss Function Equation:

$$Loss = a * MSE(X, \hat{X}) + (1 - a) * CosineSimilarity(X, \hat{X})$$

Where:

- $X$  represents the original embeddings,
- $\hat{X}$  represents the reconstructed embeddings,
- $\alpha$  balances the importance between reconstruction fidelity and semantic preservation.

## Step 2: Visualize Embeddings

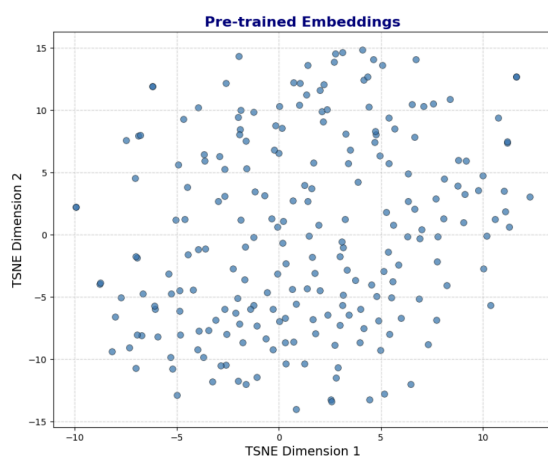


Figure 2 : Pre trained embeddings

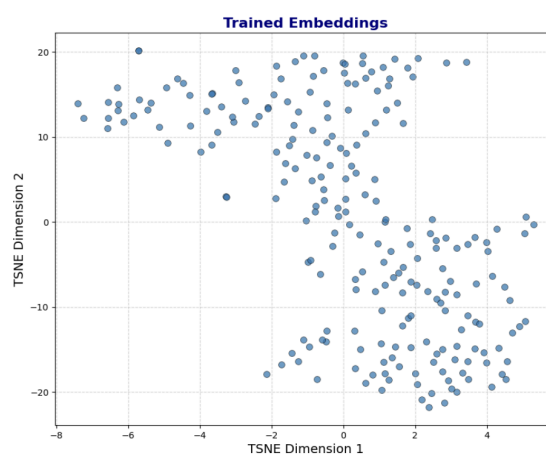


Figure 3 : Trained embeddings

```
Clustering Evaluation for Pre-trained Embeddings:
- Silhouette Score (higher is better): 0.3596
- Calinski-Harabasz Index (higher is better): 221.7845
- Davies-Bouldin Index (lower is better): 0.8613

-----


Clustering Evaluation for Trained Embeddings:
- Silhouette Score (higher is better): 0.4675
- Calinski-Harabasz Index (higher is better): 909.8960
- Davies-Bouldin Index (lower is better): 0.7175
```

In Step 2, the training process was evaluated by comparing clustering metrics and visualizing the embeddings before and after training. The trained embeddings exhibited clearer semantic distinctions among clusters, as evidenced by both the quantitative metrics and the visual distribution. In the pre-trained embeddings, shown in the first plot, the points are more dispersed with less defined groupings, which reflects a lack of clear thematic separation. In the trained embeddings plot, while distinct clusters are not sharply separated, there are more defined groupings in specific areas: notably in the upper-left corner, lower-right corner, and central region

Quantitatively, this improvement is reflected in the clustering metrics. The Silhouette Score increased from 0.3596 to 0.4675, indicating better-defined and more distinct clusters. The Calinski-Harabasz Index, which evaluates compactness and separation, rose significantly from 221.7845 to 909.8960, suggesting tighter, more defined clusters. The Davies-Bouldin Index, which assesses cluster overlap, decreased from 0.8613 to 0.7175, further confirming improved separation between themes. Together, these results show that the training process helped the model better capture semantic nuances across Paul Graham's essays.

## Step 3: Testing the search system

Results:

 Top 5 results for 'users':

- 
1. Chapter 1 of Ansi Common Lisp - Similarity: 0.9851
  2. Chapter 2 of Ansi Common Lisp - Similarity: 0.9851
  3. Lisp for Web-Based Applications - Similarity: -0.1431
  4. What Made Lisp Different - Similarity: -0.1475
  5. Programming Bottom-Up - Similarity: -0.1612

 Top 5 results for 'love':


- 
1. Chapter 2 of Ansi Common Lisp - Similarity: 0.9361
  2. Chapter 1 of Ansi Common Lisp - Similarity: 0.9361
  3. What Made Lisp Different - Similarity: -0.0856

4. Lisp for Web-Based Applications - Similarity: -0.0918

5. Programming Bottom-Up - Similarity: -0.0957

-----

 **Test with a non-obvious result for query 'writing':**

 Top 5 results for 'writing':

-----

1. Chapter 1 of Ansi Common Lisp - Similarity: 0.886

2. Chapter 2 of Ansi Common Lisp - Similarity: 0.886

3. What Made Lisp Different - Similarity: 0.1404

4. Programming Bottom-Up - Similarity: 0.1287

5. Lisp for Web-Based Applications - Similarity: 0.1281

### **Conclusion:**

The results of the search tests indicate a limitation in the model's ability to differentiate effectively between distinct queries. Despite varied inputs—such as "users," "love," and "writing"—the search results show significant overlap, with the top-ranked documents often being the same or similar across different queries. This pattern suggests that the model may be overemphasizing general semantic similarities, potentially due to recurring themes and overlapping vocabulary in Paul Graham's essays. The clustering analysis further illustrates this: while trained embeddings exhibit more defined clusters than the pre-trained ones, clear separation by topic is still lacking, with noticeable but vague groupings in certain areas.

To address the current model's limitations, switching to a model like sBERT, optimized for sentence similarity, would likely improve semantic alignment and capture finer distinctions across essays. Additionally, integrating contrastive loss during training could help separate embeddings for thematically similar yet distinct content, reducing overlap and improving query-specific retrieval accuracy. These adjustments target both semantic depth and document differentiation, addressing key limitations observed in the results.