Songs semantic similarity

User will provide a search query and the system shall provide the similar song details based on comparing between the search query and songs lyrics.

Data preprocessing

- Convert lyrics text to lower case.
- Removing punctuations and special characters.
  Tokenize text into words.
- Remove stop words that in English and not add to the overall meaning.
- Stemmer the words to make it back to the basic form ex: running to run.

Sentence embeddings

The choice was sentence embedding mpnet model from Hugging face it is a powerful aproch for getting the meaning of the lyrics not like other techniques like word2vec which behave with every word independently and not capture the overall sentence meaning.

1. Building Block - Transformer Encoder

At its core, the MPNet model leverages the transformer encoder , it is consists of main sub-modules that work together to understand the meaning of a sentence:

- **Positional Encoding:**

  adds information about where each word appears in a sentence, This helps the model understand sentence structure and meaning, All-mpnet-base-v2 uses this technique to process language effectively.

- **Self-Attention Layer:**
  This layer analyzes the relationships between words within a sentence. It is like each word has an "attention score" assigned to other words based on their relevance in the context. The self-attention layer allows the model to understand how each word connects to others and contributes to the overall meaning.

- **Feed Forward Layer:**
  This layer adds non-linearity to the model. Unlike the self-attention layer that focuses on relationships, the feed forward layer allows the model to capture more complex patterns and dependencies within the sentence.

2. Multi-headed Attention (MHA)

- One key difference between MPNet and standard transformers is the use of Multi-headed Attention (MHA). Here's what this means:
- A standard transformer encoder typically has a single self-attention layer but MPNet, however, employs multiple "heads" within its self-attention layer. Each head focuses on attending to different aspects of the relationships between words simultaneously.
- It is like having multiple analysts examining the same sentence, each focusing on a specific angle (e.g., grammatical structure, word order, thematic connections). This allows MPNet to capture a richer understanding of the sentence's meaning.

3. Masked and Permuted Language Modeling (MPNet)

The MPNet stands for Masked and Permuted Language Modeling. These techniques go beyond the standard training methods used for transformers:

- **Masked Language Modeling (MLM):**
  This technique is common in transformers. It involves training the model to predict a masked word based on the surrounding context. This helps the model understand the meaning and relationships between words within the sentence.
- **Permutation Invariance:**
  MPNet additionally trains the model to handle permutations of words within a sentence. This means the model should be able to understand the meaning regardless of word order, This is particularly beneficial for languages with more flexible word order compared to English.

4. The pooling layer

Receives the output from the Transformer model. This output is likely a series of vectors, one for each word in the input sentence. Each word vector captures the semantic meaning of that specific word within the context of the sentence.

- **Pooling Mode - Mean Tokens:**

- It takes the average of all the individual word vectors generated by the transformer model. Imagine each word vector as a point in a high-dimensional space, averaging these points essentially creates a new central point that represents the "average meaning" of all the words in the sentence.
- Mean pooling offers a robust way to capture the overall semantic meaning of a sentence.
  By averaging all word vectors, it incorporates information from every word while mitigating the influence of any single word.
- After applying the mean_tokens pooling mode, the layer generates a single 768-dimension vector representing the entire sentence. This vector encodes the semantic meaning extracted by the transformer model.
- We can use the same SentenceTransformer model to generate similar vectors for song lyrics.

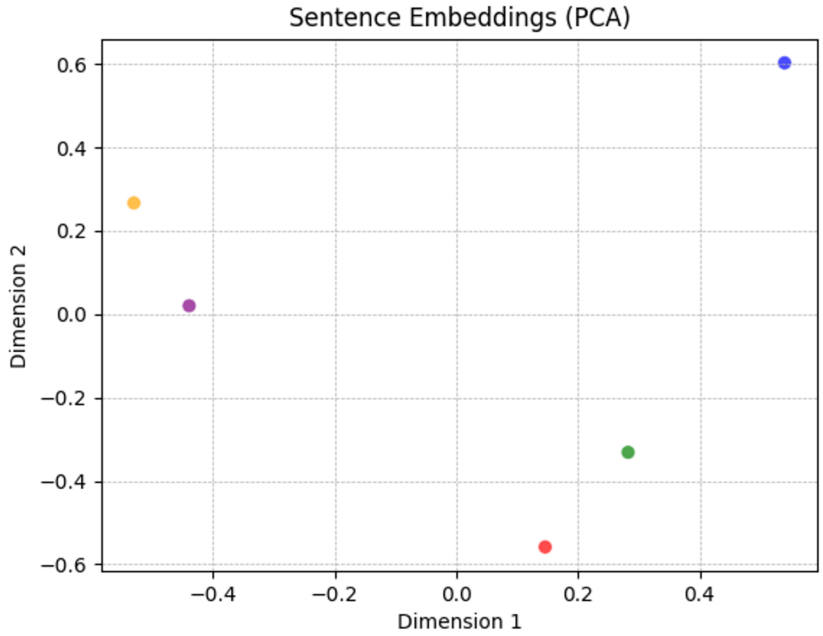
5. The normalization layer

Receives the 768-dimension vector generated by the pooling layer. This vector encodes the semantic meaning of the sentence but might not be in a directly comparable format.

- **Normalization Technique:**
  - The normalization layer applies a specific technique to transform the input vector by L2 normalization.
  - L2 normalization is a common technique used to ensure all vectors have a unit length (Euclidean norm of 1). Imagine the vector as an arrow in a high-dimensional space
  - Normalization essentially adjusts the length of this arrow to 1, making its magnitude overall strength consistent.
- **Normalization offers several advantages for semantic search:**
  - Fairer Comparisons: By having all sentence and song lyric vectors on the same scale (unit length), cosine similarity (or other similarity metrics) can focus purely on the directional differences between vectors. This leads to fairer comparisons based on semantic similarity rather than the original vector magnitudes.
  - Improved Numerical Stability: Normalization can improve the stability of calculations, especially with complex similarity metrics.

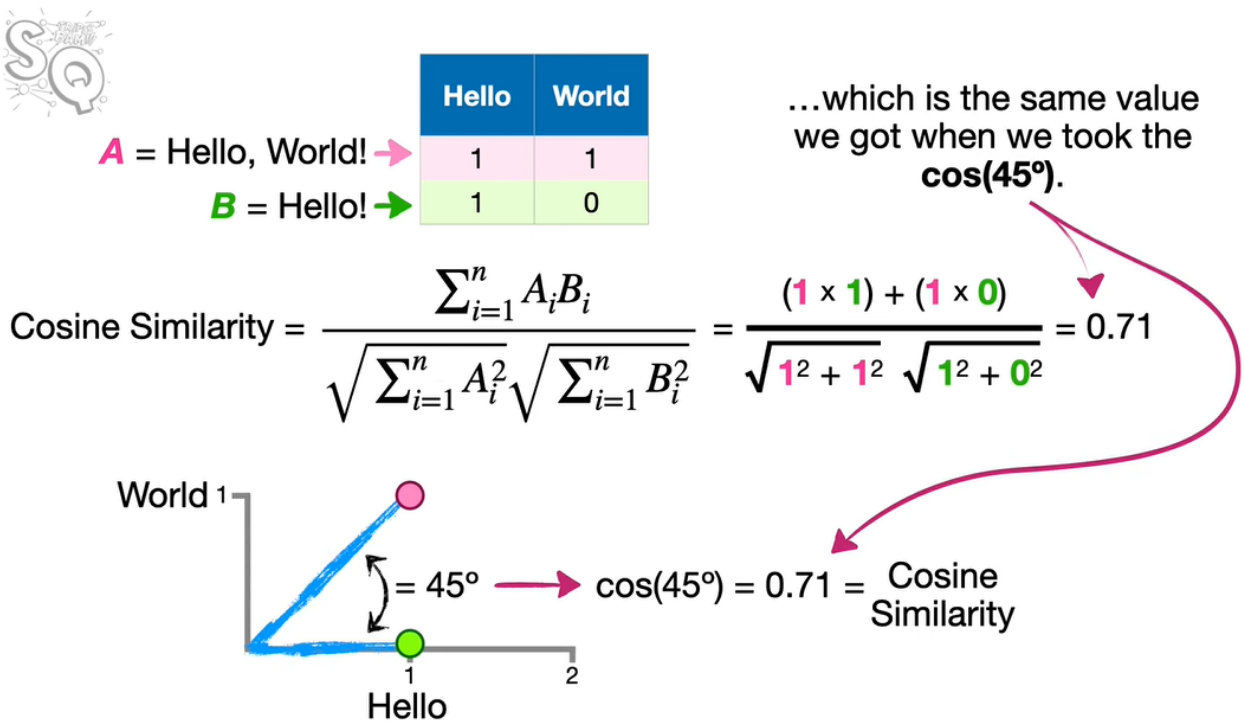
- Standardized Space: It establishes a consistent space for vector comparisons, simplifying the overall process.

If we apply the model mpnet on 5 songs and use PCA to reduce dimensionality and plot the space:

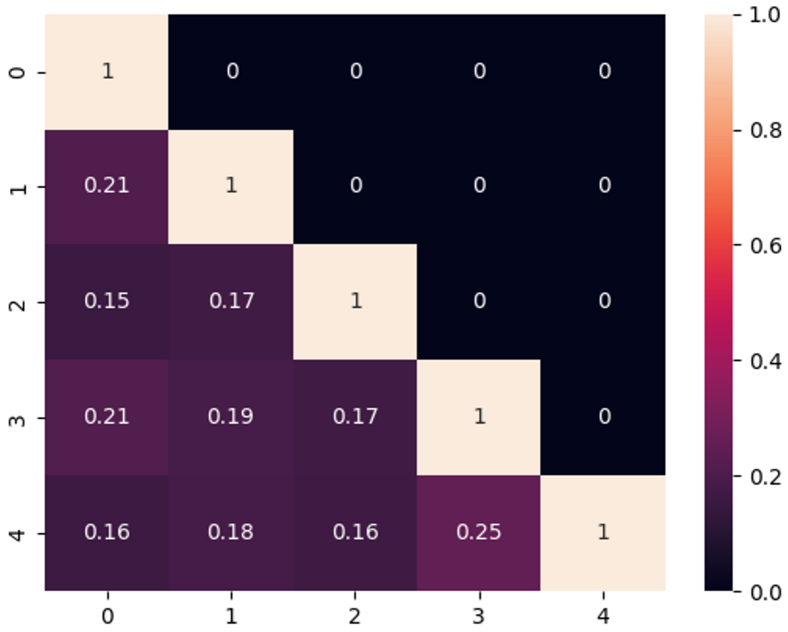


Cosine Similarity: Measuring Directional Similarity

Cosine similarity is a metric used to measure the similarity between two vectors in a multidimensional space. It specifically focuses on the directional similarity, meaning it captures how closely the vectors are aligned, regardless of their magnitudes (lengths). This makes it particularly suitable for comparing high-dimensional vectors like those generated by sentence transformers, which often have similar magnitudes due to normalization.



By taking the first 5 songs and apply our techniques:



By these techniques our songs semantic search microservice is ready lets use streamlit to try it:

the song is Hotline Bling by Bille Elish and the real line is:

“I know the hotline bling and that can only mean one thing”