```
pip install sentence-transformers scikit-learn
```

```
Requirement already satisfied: sentence-transformers in /usr/local/lib/python3.10/dist-packages (3.3.1)
Requirement already satisfied: scikit-learn in /usr/local/lib/python3.10/dist-packages (1.6.0)
Requirement already satisfied: transformers<5.0.0,>=4.41.0 in /usr/local/lib/python3.10/dist-packages (from sentence-transformers)
Requirement already satisfied: tqdm in /usr/local/lib/python3.10/dist-packages (from sentence-transformers) (4.67.1)
Requirement already satisfied: torch>=1.11.0 in /usr/local/lib/python3.10/dist-packages (from sentence-transformers) (2.5.1+cu121)
Requirement already satisfied: scipy in /usr/local/lib/python3.10/dist-packages (from sentence-transformers) (1.13.1)
Requirement already satisfied: huggingface-hub>=0.20.0 in /usr/local/lib/python3.10/dist-packages (from sentence-transformers) (0.27
Requirement already satisfied: Pillow in /usr/local/lib/python3.10/dist-packages (from sentence-transformers) (11.0.0)
Requirement already satisfied: numpy>=1.19.5 in /usr/local/lib/python3.10/dist-packages (from scikit-learn) (1.26.4)
Requirement already satisfied: joblib>=1.2.0 in /usr/local/lib/python3.10/dist-packages (from scikit-learn) (1.4.2)
Requirement already satisfied: threadpoolctl>=3.1.0 in /usr/local/lib/python3.10/dist-packages (from scikit-learn) (3.5.0)
Requirement already satisfied: filelock in /usr/local/lib/python3.10/dist-packages (from huggingface-hub>=0.20.0->sentence-transform
Requirement already satisfied: fsspec>=2023.5.0 in /usr/local/lib/python3.10/dist-packages (from huggingface-hub>=0.20.0->sentence-t
Requirement already satisfied: packaging>=20.9 in /usr/local/lib/python3.10/dist-packages (from huggingface-hub>=0.20.0->sentence-tr
Requirement already satisfied: pyyaml>=5.1 in /usr/local/lib/python3.10/dist-packages (from huggingface-hub>=0.20.0->sentence-transf
Requirement already satisfied: requests in /usr/local/lib/python3.10/dist-packages (from huggingface-hub>=0.20.0->sentence-transform
Requirement already satisfied: typing-extensions>=3.7.4.3 in /usr/local/lib/python3.10/dist-packages (from huggingface-hub>=0.20.0->
Requirement already satisfied: networkx in /usr/local/lib/python3.10/dist-packages (from torch>=1.11.0->sentence-transformers) (3.4
Requirement already satisfied: jinja2 in /usr/local/lib/python3.10/dist-packages (from torch>=1.11.0->sentence-transformers) (3.1.4
Requirement already satisfied: sympy==1.13.1 in /usr/local/lib/python3.10/dist-packages (from torch>=1.11.0->sentence-transformers)
Requirement already satisfied: mpmath<1.4,>=1.1.0 in /usr/local/lib/python3.10/dist-packages (from sympy==1.13.1->torch>=1.11.0->sen
Requirement already satisfied: regex!=2019.12.17 in /usr/local/lib/python3.10/dist-packages (from transformers<5.0.0,>=4.41.0->sente
Requirement already satisfied: tokenizers<0.22,>=0.21 in /usr/local/lib/python3.10/dist-packages (from transformers<5.0.0,>=4.41.0->
Requirement already satisfied: safetensors>=0.4.1 in /usr/local/lib/python3.10/dist-packages (from transformers<5.0.0,>=4.41.0->sent
Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python3.10/dist-packages (from jinja2->torch>=1.11.0->sentence-tran
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from requests->huggingface-hub>=
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests->huggingface-hub>=0.20.0->sent
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from requests->huggingface-hub>=0.20.0
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from requests->huggingface-hub>=0.20.0
```

```
from sentence_transformers import SentenceTransformer
from sklearn.metrics.pairwise import cosine_similarity
import numpy as np

def reorder_paragraphs(paragraphs):
    # Step 1: Load the embedding model
    model = SentenceTransformer('all-MiniLM-L6-v2')  # Use a lightweight sentence embedding model

    # Step 2: Generate embeddings for each paragraph
    embeddings = model.encode(paragraphs)

    # Step 3: Compute cosine similarity matrix
    similarity_matrix = cosine_similarity(embeddings)

    # Step 4: Rank paragraphs based on similarity to others (e.g., to the first paragraph or as a cluster)
    scores = similarity_matrix.sum(axis=1)  # Aggregate similarity scores for each paragraph
    ordered_indices = np.argsort(scores)[::-1]  # Sort indices by descending similarity score

    # Step 5: Reorder paragraphs
    reordered_paragraphs = [paragraphs[i] for i in ordered_indices]

    return reordered_paragraphs

# Example usage
paragraphs = [
    "They believed the whispers carried the secrets of the forest and warnings of danger.",
    "Though many dismissed it as folklore, the woods remained untraveled, their dense canopy casting an eternal twilight that both intrig
    "Deep in the heart of a forgotten forest, there lay a village surrounded by towering trees and shrouded in mystery. The villagers spo
]

reordered = reorder_paragraphs(paragraphs)
for i, para in enumerate(reordered, 1):
    print(f"Paragraph {i}: {para}")
```

```
Paragraph 1: Deep in the heart of a forgotten forest, there lay a village surrounded by towering trees and shrouded in mystery. The
Paragraph 2: They believed the whispers carried the secrets of the forest and warnings of danger.
Paragraph 3: Though many dismissed it as folklore, the woods remained untraveled, their dense canopy casting an eternal twilight tha
```