Question-Answering and Summarization Pipeline Documentation

Introduction

This documentation outlines the core functionality of an NLP (Natural Language Processing) pipeline designed for question-answering (QA) and summarization. The primary goal of this pipeline is to select the top 3 relevant sentences from a set of documents and generate concise summaries.

Core Flow

1. **Initialization**
   - Device Configuration: Detects GPU availability and configures the execution device (CPU or GPU).
   - Model and Tokenizer Initialization: Utilizes BERT for sentence classification and loads pre-trained models and tokenizers for BART and PEGASUS.
   - DPR Models: Initializes question and context encoders from Facebook's DPR (Deep Passage Retrieval) models.
2. **Document Retrieval and Preprocessing**
   - Iterating Through Data Points: Iterates through a dataset of passages, queries, and answers.
   - Query Processing: Tokenizes and encodes user queries, generating question embeddings with the DPR question encoder.
   - Document Scoring: Calculates cosine similarity between query embeddings and document embeddings to identify relevant passages.
   - Selecting Relevant Passages: Sorts passages by similarity and selects the top 3 most relevant ones.
3. **Summarization**
   - Utilizes the `SentenceGraphPruner` class to prune redundant sentences based on embedding similarity.
   - Employs the `MMR` class to generate a summary using the Maximal Marginal Relevance algorithm.
   - Provides abstractive summaries using BART and PEGASUS models.
   - Selects the most diverse summary as the final summary
4. **Answer Generation**
   - Iterates through selected and summarized passages.
   - Generates answers based on the processed passages.
   - Stores generated answers for presentation or further analysis.

Summarization Function

SentenceGraphPruner Class Overview

- Prunes redundant sentences based on embedding similarity.

- Uses SentenceTransformer for sentence embeddings and spaCy for tokenization.

MMR (Maximal Marginal Relevance) Class Overview

- Generates summaries using the Maximal Marginal Relevance algorithm.
- Utilizes TF-IDF vectors and cosine similarity for sentence scoring.

Abstractive Summary Methodology Overview

- Provides abstractive summaries using BART and PEGASUS models.

Pipeline Function

- Executes a multi-step process to generate summaries.
- Includes graph pruning, MMR extraction, abstractive summarization, diversity calculation, and selection of the most diverse summary.

Conclusion

This NLP pipeline combines document retrieval, summarization, and answer generation to deliver concise and relevant answers to user queries. Its core flow revolves around selecting the top 3 relevant passages and summarizing them effectively.