**AI Research Paper Summarization System**

**1. Project Overview**

**1.1 Introduction**

With the increasing volume of research papers published daily, it becomes challenging for researchers to extract key insights efficiently. Manual reading is time-consuming and impractical at scale. Therefore, automating the summarization process using Natural Language Processing (NLP) techniques can significantly enhance the productivity of researchers and academicians.

This project aims to develop an AI-powered summarization system that automatically extracts and summarizes research papers from ArXiv.org and other scientific sources. The system incorporates two complementary types of summarization:

* **Extractive Summarization (TextRank)**: Identifies and ranks the most important sentences from the original text based on relevance and importance.
* **Abstractive Summarization (T5 Transformer Model)**: Generates human-like summaries by rephrasing and paraphrasing the original content.

Including both types of summarization provides users with comparative insight. Extractive summarization offers precise, factual content directly from the source, while abstractive summarization provides a more natural and readable summary. This dual approach allows for benchmarking the quality of generated summaries through evaluation metrics such as ROUGE, and enhances transparency by enabling users to see the raw and synthesized summaries side by side.

**1.2 Objectives**

* Automate the summarization of full-length research papers.
* Provide two types of summaries: extractive (key sentence extraction) and abstractive (human-like summaries).
* Ensure high-quality outputs by optimizing pre-processing, summarization, and evaluation techniques.
* Deploy a user-friendly web application using Streamlit for real-time summarization.

**2. Methodology**

**2.1 System Architecture**

The system follows a pipeline-based approach with the following components:

1. Data Extraction: Scraping research papers from ArXiv.org.
2. Preprocessing: Cleaning text by removing references, symbols, numbers, and metadata.
3. Extractive Summarization: Applying TextRank to extract key sentences.
4. Abstractive Summarization: Using a T5 Transformer Model to generate human-like summaries.
5. Evaluation: Using ROUGE Score to compare summarization quality.
6. Deployment: Deploying the system as a web application on Streamlit Cloud.

**2.2 Technology Stack**

| **Component** | **Technology Used** |
| --- | --- |
| Web Framework | Streamlit |
| Text Extraction | pdfplumber |
| Extractive Summarization | TextRank (TF-IDF & PageRank) |
| Abstractive Summarization | T5 Transformer (Hugging Face) |
| Model Training | PyTorch |
| Evaluation Metrics | ROUGE Score |
| Deployment | Streamlit Cloud |
| Version Control | GitHub |

**3. Implementation**

**3.1 Data Collection**

Research papers are fetched from ArXiv.org and stored in PDF format. The PDFs are then processed to extract readable text.

**3.2 Text Preprocessing**

Before summarization, text is cleaned and preprocessed to ensure quality input:

* Removal of author names, affiliations, and references (e.g., [1], [crossref]).
* Elimination of special characters, numbers, and excessive punctuation.
* Normalization of text while preserving sentence structure.

**3.3 Extractive Summarization (TextRank)**

TextRank is an unsupervised graph-based algorithm that ranks sentences based on importance:

1. Convert text into TF-IDF vectors.
2. Compute sentence similarity using cosine similarity.
3. Build a graph where nodes are sentences.
4. Apply PageRank to identify the most important sentences.
5. Extract the top-ranked sentences as the summary.

**3.4 Abstractive Summarization (T5 Transformer Model)**

Abstractive summarization involves generating new text instead of extracting key sentences. The T5 model is used for this purpose:

* Fine-tuned on scientific data.
* Uses beam search and penalty adjustments to reduce repetition.
* Generates human-like concise summaries.

**3.5 Evaluation (ROUGE Score)**

To measure summarization quality, ROUGE-1 and ROUGE-2 Scores are computed:

* ROUGE-1 measures unigram (word-level) overlap.
* ROUGE-2 measures bigram (two-word sequence) overlap.

**4. Deployment**

The final system is deployed as a web application using Streamlit Cloud:

* Allows users to upload PDFs.
* Generates extractive and abstractive summaries in real time.
* Displays ROUGE Scores for quality evaluation.
* Publicly accessible via Streamlit Cloud URL.

**5. Challenges and Solutions**

**5.1 Poor Text Extraction**

* Issue: Extracted text contained unwanted metadata.
* Solution: Applied advanced text cleaning to remove headers, citations, numbers, and unwanted symbols.

**5.2 Repetitive Abstractive Summaries**

* Issue: The T5 model generated repeated words.
* Solution: Adjusted temperature, repetition penalty, and beam search settings.

**5.3 Deployment Errors on Streamlit Cloud**

* Issue: The sentencepiece library was missing during deployment.
* Solution: Added sentencepiece to requirements.txt and redeployed.

**6. Results and Findings**

* Extractive Summaries (TextRank) produced concise and factually correct key sentences.
* Abstractive Summaries (T5) generated fluent, human-like summaries.
* ROUGE Scores showed that abstractive summaries had strong word and phrase overlap with extractive summaries.
* The web application successfully provided on-the-fly summarization and evaluation.

**7. Conclusion and Future Enhancements**

**7.1 Conclusion**

This project successfully demonstrates an AI-driven research paper summarization system that extracts, processes, and summarizes scientific articles. It is deployed as a user-friendly web application, allowing real-time summarization and evaluation.

**7.2 Future Enhancements**

* Fine-tuning the T5 model on a domain-specific dataset such as PubMed for medical research.
* Adding support for multi-document summarization.
* Integrating advanced summarization models such as Pegasus or BART for better performance.

**8. References**

1. Google Research, "Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer," 2020.
2. Mihalcea, R. and Tarau, P., "TextRank: Bringing Order into Texts," 2004.
3. ArXiv API Documentation: <https://arxiv.org/help/api/index>

**9. Final Deployed Web Application**

URL: https://ai-research-paper-summarizer-bddthsqmmgjugwk7w8wqy9.streamlit.app/