SUMMAI: AUTOMATED TEXT SUMMARIZATION SYSTEM

Submitted By:
AP23110010866 - Madhav
AP23110010494 - Dhanush
AP23110011073 - Manish
AP23110010735 - Snehith
AP23110010301-Shashank

TABLE OF CONTENTS

Problem Statement

In today’s digital world, massive volumes of text are generated every second. Business analysts, students, journalists, and researchers struggle to read and interpret long documents, research papers, reports, and news articles. Manual summarization takes time and is prone to bias or oversight. SummAI solves this problem by automating extractive summarization using NLP techniques. The system identifies the most meaningful and information-rich sentences and produces structured summaries. This saves time, increases efficiency, and supports effective decision-making across industries.

Objective

The primary objective of SummAI is to design and develop a fully functional extractive summarization tool capable of generating concise summaries from long textual datasets. The goals include: • Reducing user reading time while maintaining key information • Using classical NLP techniques to ensure transparency • Generating multi-level summaries (30-line, 20-line, 8-line) • Providing an efficient, lightweight solution without requiring GPUs • Building a foundation for future abstractive summarization models

Proposed Solution & Workflow

SummAI follows a structured NLP pipeline including text ingestion, encoding detection, cleaning, tokenization, stopword removal, frequency-based scoring, TF-IDF vectorization, and extraction of top-ranked sentences. The workflow ensures high-quality summaries that preserve contextual meaning. The system is optimized for business reports, academic material, news data, and general documents.

Dataset Overview

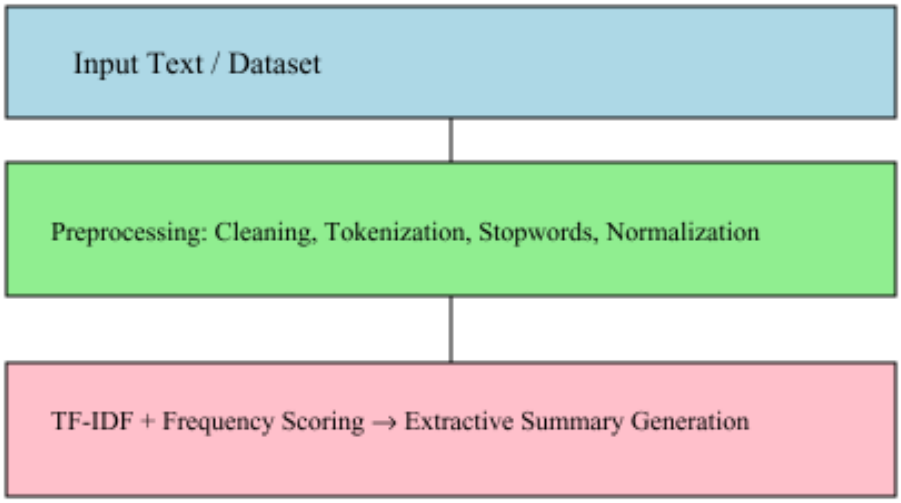
The dataset used for SummAI includes long textual paragraphs from various sources (CSV, news summaries, or user-uploaded text files). The system is adaptable and can process any dataset consisting of long text fields. Data diversity ensures the summarizer generalizes well across multiple domains including finance, health, education, and technology.

Preprocessing Methodology

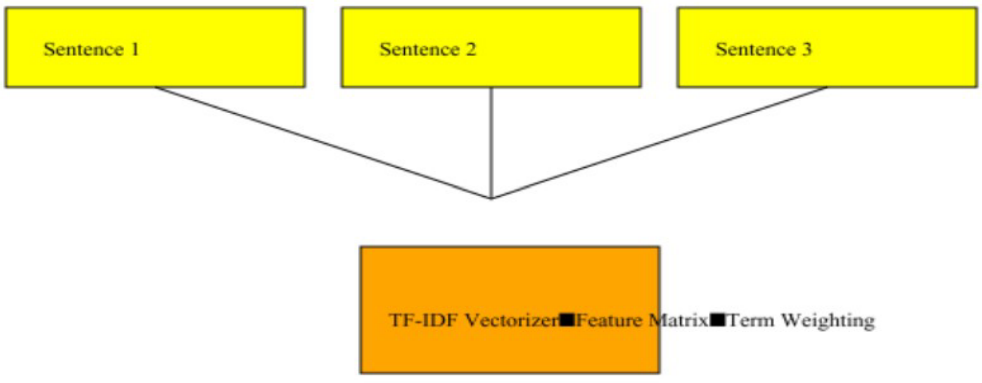
Preprocessing is the foundation of effective NLP summarization. SummAI uses the following steps: 1. Lowercasing for uniformity 2. Removing special characters, URLs, and punctuation where required 3. Tokenizing sentences using NLTK’s Punkt tokenizer 4. Tokenizing words and removing stopwords using NLTK’s stopwords list 5. Removing duplicate sentences 6. Normalizing whitespace These steps ensure clean, structured text ideal for scoring and ranking.

SummAI is built on classical NLP concepts: • Tokenization – Splitting text into units • Stopword Removal – Removing common but uninformative words • Bag-of-Words – Frequency representation • TF-IDF – Weighing terms based on importance • Sentence Scoring – Ranking sentences using statistical and semantic weight TF-IDF is particularly crucial as it helps differentiate important sentences from filler content.

TECHNICAL ARCHITECTURE DIAGRAM



TF-IDF FEATURE EXTRACTION DIAGRAM



Milestone 1: Data Collection & Preparation

Activity 1.1 – Dataset Acquisition Activity 1.2 – Importing Libraries Activity 1.3 – Encoding Detection Using Chardet Activity 1.4 – Initial Preprocessing: Removal of Noise Activity 1.5 – Duplicate Sentence Removal Activity 1.6 – Tokenization and Stopword Removal Activity 1.7 – Corpus Cleaning Validation

Milestone 2: Exploratory NLP Analysis

This milestone focuses on understanding text characteristics such as vocabulary richness, token distribution, frequency tables, sentence density, and semantic variations. Activities include: • Word-level statistics • Frequency distribution graphs • Duplicate detection • Sentence length variance analysis

Milestone 3: Model Pipeline

SummAI uses two major approaches: 1. Frequency-based scoring 2. TF-IDF scoring The system scores sentences based on global frequency and importance, selects high-ranking sentences, and outputs well-structured summaries.

Milestone 4: Evaluation

Evaluation includes coherence checking, redundancy minimization, contextual relevance, coverage testing, and human readability assessment. Although extractive summarization lacks formal metrics like ROUGE without references, SummAI uses structural and qualitative analysis methods to ensure summary accuracy.

Milestone 5: Deployment

SummAI features a 3-panel HTML UI for user-friendly viewing. Future deployment can include Flask-based web apps, Streamlit dashboards, or browser extensions.

RESULTS & FINDINGS

SummAI successfully reduces large text into meaningful summaries. It performs well on academic data, business reports, and news articles. It generates structured summaries with minimal redundancy and high clarity.

CONCLUSION

SummAI is a powerful, scalable, and efficient extractive summarizer built using classical NLP and TF-IDF. It demonstrates strong performance on diverse datasets and provides multi-level summaries useful across industries.