CECS Capstone Project
Group: Acato 1
Sage Gray
Nathan Ebbs
Cameron Kerley
Jacob Randles
Proposal Development
2/24/2025

Statement of Problem

Acato receives many solicitations, or proposals, for government contracts that could use their expertise. However, these proposals are very large, and the staff does not have the time to individually read them all to figure out which proposals would be a good fit. That is where we come in. We are tasked to automate the process of summarizing these proposals, so that Acato can know which proposals are good fits for their company and within their area of expertise and resources.

Objectives

- Develop a system to efficiently summarize large government contract proposals, allowing Acato to quickly assess their relevance.
- Utilize Large Language Models (LLMs) to generate executive summaries of lengthy documents, ensuring the most relevant details are extracted.
- Identify and evaluate models with large context windows (e.g., Claude 3.5 Sonnet, GPT-01, Llama 3.2) to handle 75–100 page documents effectively.
- Define structured instructions for summaries, including formatting, organization, and the retention of industry-specific language.
- Use metrics such as ROUGE, METEOR, and BERTScore to assess and compare summary quality across different models.
- Explore rule-based and extractive approaches, such as NLTK with sumy's TextRank, to provide non-LLM summarization options that do not require external API calls.
- Design the system to process multiple proposals quickly while minimizing computational costs and maintaining high-quality outputs.

Potential Additional Milestones

- Create an automatic method to retrieve solicitations directly from trusted websites to process almost without human input.

Methodology

Method 1: Summarization with a LLM

One approach to build the Rapid Identification and Vetting system(RIV) involves using modern AI technology for Natural Language Processing. This approach involves processing documents with a Large Language Model to extract essential information and generating an executive summary. The first step in this process is selecting a model suitable for large document summarization. Large Language Models like chatGPT use something called a context window to define how long an input can be. Our system will need the largest context window possible for 75 to 100 page documents. When a model ingests text like a document, it converts words to tokens, and on average one word becomes two to three tokens. To further extrapolate that average, a double spaced 100 page document averages around 25,000 words which is between 50,000 - 75,000 tokens. This one requirement filters the list of commercially available LLM's down to 3 or 4 companies. Anthropic AI's Claude 3.5 Sonnet with a 200,000 token context window or OpenAI's GPT-o1 and Meta's Llama 3.2 both with a 128,000 token context window(Cole).

To evaluate the available models and APIs, we need to establish a set of instructions that define an effective summary within the context of Acatos' business capabilities. These instructions will outline:

1. The required format for the document summary.
2. The approach to summarization, such as structuring it by header or by page.
3. Additional guidelines, such as maintaining organization-specific language.

Additionally, we need a standardized method to evaluate summaries produced by different models. Various metrics, including ROUGE, METEOR, and BERTScore, are commonly used to assess LLM-generated summaries, though each comes with its own challenges(Bais). The ROUGE and METEOR metrics will require a human written summary of test documents for comparison and BERTScores become computationally more expensive for longer documents. With the best model identified we can begin assessing new document summaries for potential issues.

Method 2: NLTK Library with sumy's TextRank

The Natural Language Toolkit for python contains many tools for parsing text and extracting the highest valued keywords and phrases from it. The sumy library contains TextRank, an unsupervised machine learning algorithm that's a special take on PageRank, the algorithm Google uses to sort through webpages. Since TextRank does not require training data, it is able to summarize content based on weights provided by the user instead of needing to train on the limited amount of data we've been provided. This method could also be expanded using BERT, a transformer for extractive text summarization. We are planning on implementing this as a simple python script that will ask the user for the location of the document and whether it is a word document or a PDF and how many sentences the summary should be, then run the code to fit the input type. It will output a different amount of information depending on the number of sentences asked for by the user, but will always use the TextRank algorithm to decide the most pertinent information. This method is less efficient than using a large language model, but would require no API calls to outside servers. To make it

Suggested tools to build LLM summarization:

- Python:
    - The anthropic API for Claude 3.5.
    - The OpenAPI for ChatGPT and GPT-o1.
    - The llamaapi for Meta's Llama models.
    - The sentence-transformers library for BERTScore similarity calculations.

Suggested tools to build NLTK summarization:

- Python
    - NLTK (Natural Language Toolkit) or BeautifulSoup libraries to process words in a document
    - Pandas to store document embeddings in dataframes.
    - Sklearn and scipy will be used for feature extraction and CountVectorizer
    - Sumy contains TextRank, an algorithm for summarizing text

Timeline

- Week 5: Separate the document archive into training and test datasets using a document parser.

- Week 6: Start evaluating summaries generated by LLM's and NLTK by using one of the three summary metrics.
- Week 7: Determine what model and API create the best summary for Acato business leads.
- Week 8: Evaluate NLTK and LLM summaries against test documents to identify potential bugs or variations in summary quality.
- Week 9-13: Iterate on the current system by fixing bugs and testing accuracy while receiving client and advisor feedback.

Deliverables

- The best LLM for Acato to use in their company that will be used to summarize the documents into a usable and digestible format.
- A program to take the summaries and process them into a number that indicates whether it is a good match for the company to take on.

Citations:

Bais, Gourav. "LLM Evaluation for Text Summarization." *Neptune.Ai*, 25 Sept. 2024, neptune.ai/blog/llm-evaluation-text-summarization.

Cole. "LLMs with Largest Context Windows." *Custom Software Development and Digital Transformation For Enterprise*, Codingscape, 22 Oct. 2024, codingscape.com/blog/llms-with-largest-context-windows.