Problem Statement:

Background

With the growing demand of multiple factors, this is an era of continuous research and developments. Scientists and researchers are working hard to provide advanced technologies and methods to serve the well-being of a greater society. However, they struggle a lot when it comes to gathering relevant background information on any research topic. They need to spend a substantial amount of time to go through previous research papers for literature reviews and decide whether a particular paper is of their interest or not.

Objective

To solve the above problem, the goal of this project is to **build an efficient Knowledge Management System capable of querying and retrieval to help fellow researchers with the recommendation of research papers relevant to their topic of interest**. The system which we will build accepts research papers (pdfs) and the user query as inputs and outputs those papers which are highly relevant. We will take into consideration the information retrieval and aspect identification techniques of Natural Language Processing (NLP) to identify key aspects of chosen research papers and extract its summarized information. Similarity scores will be measured afterwards between user query and extracted data to pick the most relevant research papers as per user's interest.

Data:

The dataset for testing our NLP model will be some (belonging to diversified domains) research articles from websites like Google Scholar or Research Gate. We will then apply random sampling to select 20 research articles out of the collection.

Approach:

To solve our problem, we need to accomplish the whole task in three phases.

1) At first phase, we need to extract important information and identify the aspect of each individual research paper using text summarization provided by end users.
2) The task of second phase will be to measure the similarity score between the user query and the summarized text of every section which will in turn help to calculate average similarity score of a paper.
3) At final phase, we will build an interface to recommend end-users the most relevant research papers as per similarity scores.

Task 1:

Keywords extraction from each header along with text summarization of each section.

Method -

- Extract important words from user queries and pre-process with the help of NLTK library.
    - Word tokenization and remove less important words like stop words, punctuation using **regex** function.
    - Can also take help from **NER** or **LDA** for determination of named entities, topics. Words will be categorized into some pre-defined categories like place, item etc.
    - Use Penn Treebank POS (part of speech) tagging for automatically assigning words to relevant part of speech as per the context of sentence.

- Parse the PDF into text (using **PDF -> TXT** tools) and extract key words from title of a research paper. [pre-processing and **NER, LDA, POS tagging**]
- Remove 'References' from our data to avoid ambiguity (we will try to enhance our scope once the initial phase of research becomes successful)

- Summarize important sentences from description(body) of each section such that it is easier to derive the aspect of the research paper.

  To achieve above, we may follow below steps:

    - Preprocess the text using stop word removal, lower-case conversion, removal of special characters, punctuation etc.
    - Create a '**Term-Sentence**' matrix (using TF-IDF method) where each row corresponds to a word and each column denotes a sentence found in section's body.
    - Each entry **a_{ij}** of the matrix will be the **weightage of** each word-sentence pair.
    - Perform **SVD** (A = U∑Vt) on term-sentence matrix to project the sentence vectors into reduced latent-semantic space using singular vectors (Vt).
    - Take a threshold value of the projection i.e. **∑*Vt** (the higher the projection more relevant are that sentence) in reduced dimension and select those sentences which are above the threshold to include in summary.
- The summaries of respective sections along with keywords extracted from header are ready.
- Create a table with columns as per the name of section headers of a research paper (Abstract, Introduction, Related Work, Methods, Results, Evaluation, Conclusion) and rows will be title of different research papers.

<u>Task 2:</u>

Filter based on similarity scores of text summaries with user query.

Method -

- Convert all extracted keywords and summaries into vectors.
  [Use TF-IDF vectorization/Word2Vec on processed query, title, and summary text]
- Calculate the **Cosine Similarity** between the query vector and other vectors (**title/section summaries**) for each one of the given papers.
  [Measure individually for each of the sections and store the score in a new Table]

- Take the average of each section to measure overall similarity score of every paper and sort them in descending manner so that our system can provide recommendations (of research papers) as per the similarity.

<u>Task 3:</u>

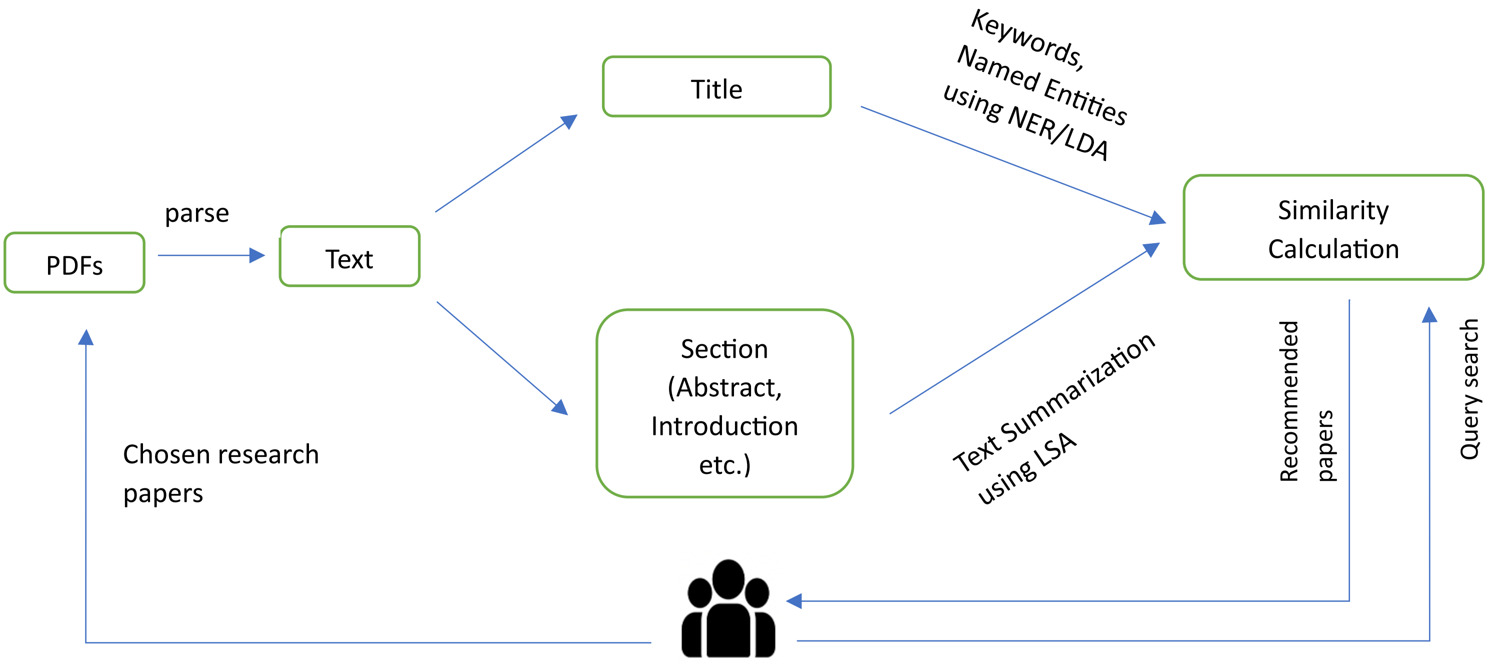
Building an interface for end-users.

Method -

- The first user input will be the research question of his/her interest -> User query
- The second user input will be the chosen research paper(s) which the user might have downloaded before -> PDFs
  - We will use tools like **PyPDF2** for converting pdfs into text files.
  - A constraint of number of input pdfs will be fixed for users such that system will not be able to proceed below that limit [For eg. Users at least need 5 pdfs to input].
  - As a future enhancement, we will also try to use the available APIs of scholar websites like **arXiv** which allow pdfs to get downloaded automatically within our system. Then users need not to explicitly provide input pdfs for analyzing.

- The user interface will be built using **Stream Lit** library.
- The output of our system will be the recommendation of Top -N relevant research papers along with their cosine similarity percentage.

Workflow Diagram:


PDFs
parse
Text
Title
Section (Abstract, Introduction etc.)
Keywords, Named Entities using NER/LDA
Text Summarization using LSA
Similarity Calculation
Recommended papers
Query search
Chosen research papers

Evaluation Process:

1) To check the overall validity of our model's output we need to depend on **human evaluation**. We will collaborate with some domain experts and request them to rank the most relevant research papers from the user given inputs in descending manner such that it helps to solve the user query. Once they provide feedback, we can validate their outputs with system generated outputs and observe the similarity between both.
2) We can use **Coherence Score** which measures how interpretable the topics are to humans for evaluating the quality of aspects.
3) Also, **ROUGE** can be used for evaluation of section summaries with respect to human generated summaries.

Exploratory Data Analysis:

1) We will create some **Word Clouds** on each section of individual research papers to check the frequency of occurrence of words.
2) We may check the **average sentence length** of each section of individual research papers to get an idea of length of sentences in Summary.

References:

- https://digitalcommons.unl.edu/cgi/viewcontent.cgi?article=1249&context=computerscidiss
- "**Text Summarization of Turkish Texts using Latent Semantic Analysis**" by Makbule Ozsoy, Ilyas Cicekli, and Ferda Alpaslan (2010)
- “**Text Summarization Techniques: A Brief Survey**” by Mehdi Allahyari, Seyedamin Pouriyeh, Mehdi Assefi, Saeid Safaei, Elizabeth D. Trippe, Juan B. Gutierrez, Krys Kochut
- “**An Unsupervised Neural Attention Model for Aspect Extraction**” by Ruidan He, Wee Sun Lee, Hwee Tou Ng, Daniel Dahlmeier
- “**An analytical study of information extraction from unstructured and multidimensional big data**” by Kiran Adnan and Rehan Akbar