SUMMPIP: UNSUPERVISED MULTI-DOCUMENT SUMMARIZATION WITH SENTENCE GRAPH COMPRESSION

GROUP MEMBERS:

1. YASH PATHAK – 2022201026
2. GAURAV KHAPEKAR - 2022201055
3. VIVEK KIRPAN – 2022201071

Team number – 8

Team name – EvolAI

Link to paper

PROBLEM STATEMENT

- State-of-the-art MDS systems are based on supervised learning, requiring relatively large amounts of labeled training data. However, obtaining training data is time consuming and resource-intensive. As a result, existing datasets are only available for limited domains.
- Despite the huge efforts of using deep neural models in summarization, they often require large-scale parallel corpora of input texts paired with their corresponding output summaries for direct supervision
- In this case, unsupervised learning approaches are appealing as they do not require labeled data for summarization.

SOLUTION PROPOSED IN PAPER

Step 1: Sentence Graph-

- Building a structured sentence graph, where the nodes correspond to the sentences from the original text and the edges are drawn based on both the lexical and the deep semantic relations between sentences.

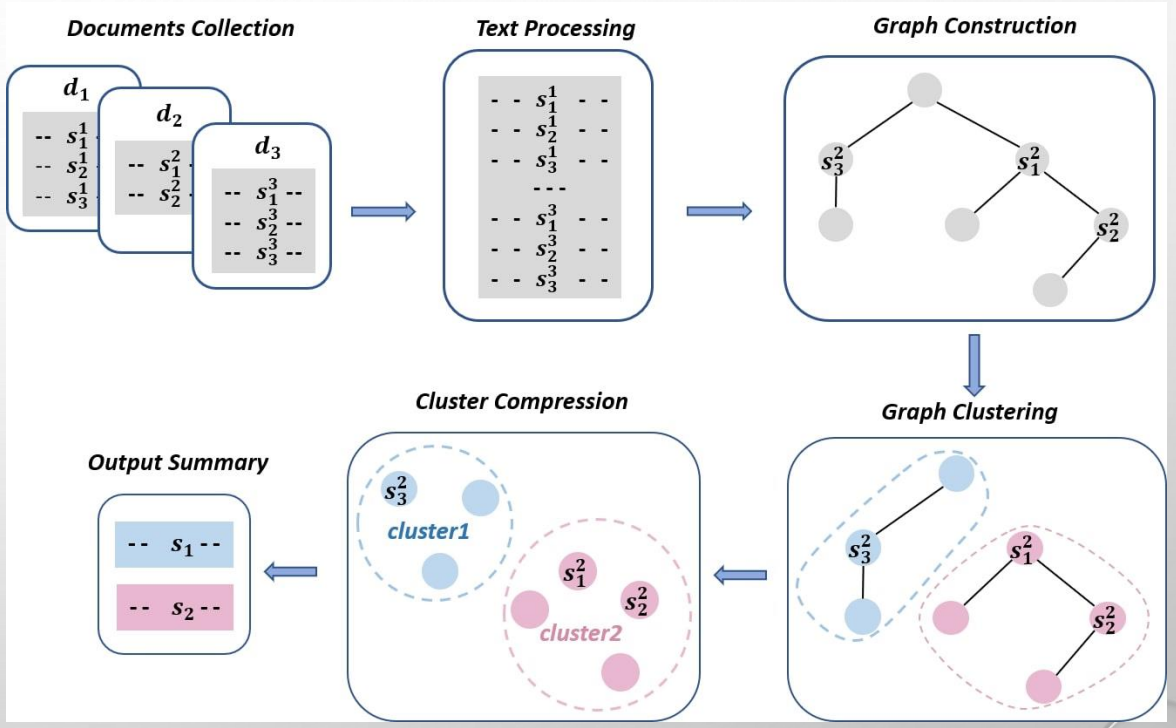


Fig. used from the paper

STEP-1: SENTENCE GRAPH

- The relation between the sentences represented by the edges is determined by checking the conditions of Deverbal Noun Reference, Discourse Markers, Entity Continuation and Sentence Similarity.
- Any of the conditions satisfying will have edge between the sentence nodes.
- The constructed sentence graph is then used for Spectral Clustering of sentences.

STEP-2: SPECTRAL CLUSTERING

- After constructing the sentence graph, spectral clustering is applied to the graph to obtain multiple clusters of sentences.
- Spectral clustering uses a mathematical matrix called the Laplacian matrix to understand the connections between sentences in a sentence graph.
- The eigenvalues and eigenvectors of the graph Laplacian matrix are computed. The eigenvectors corresponding to the smallest eigenvalues capture the low-frequency information, which corresponds to the cluster structure in the data.
- Finally, a clustering algorithm like k-means is applied to the eigenvectors corresponding to the smallest eigenvalues to partition the data points into clusters.

STEP-3: CLUSTER COMPRESSION

- The clusters now contains a set of semantic related sentences. Multi-sentence compression (MSC) generates a single summary sentence from each cluster. We get the summary by combining the single sentence from every cluster.
- The approach used is to build a word graph and take the shortest path of the words as the summary.
- This approach has been extended in this paper by considering keyphrases to adjust the compression so that the word paths with keyphrases are given higher scores.
- Then the summary with the highest score is selected as the single summary sentence of the cluster.

SCOPE OF THE PROJECT

- Summarization of Text documents using unsupervised summarization method SummPip using sentence graphs and spectral clustering.
- Implementation of the Pipeline steps of SummPip mentioned in the paper including the construction of Sentence Graph, applying Spectral Clustering and final pipeline of Cluster Compression.
- Experimenting with the implementation of SummPip and check if there is any improvement in the summary.

IMPLEMENTATION DETAILS

Implementation of the following pipelines of SummPip-

- **Sentence Graph** construction from the list of sentences of a text. Implementation of the conditions- Deverbal Noun, Discourse Markers, Entity Continuation and Sentence Semantic Similarity using Cosine Similarity between pair of sentences to determine the edge between the sentence nodes.
- **Spectral Clustering** of sentences using the sentence graph to cluster related sentences together.
- **Cluster Compression** of the clusters formed from previous step to compression the sentences of a cluster into single summary sentence. Then extracting each single sentence from all the clusters and combining them to get the final summary.

IMPLEMENTATION DETAILS

- For **Cluster Compression** we have used the **takahe** module from the paper implementation which is a huge module to implement.
- The approach used in this step is to build a word graph and pick few shorter path of words to consider for summary. Give higher scores to the word graphs containing keyphrases and pick the one with the highest score as the sentence summary of the cluster.
- Calculating the **Rouge Scores** of the generated summary with the test summary including Rouge-1, Rouge-2 and Rouge-L scores.

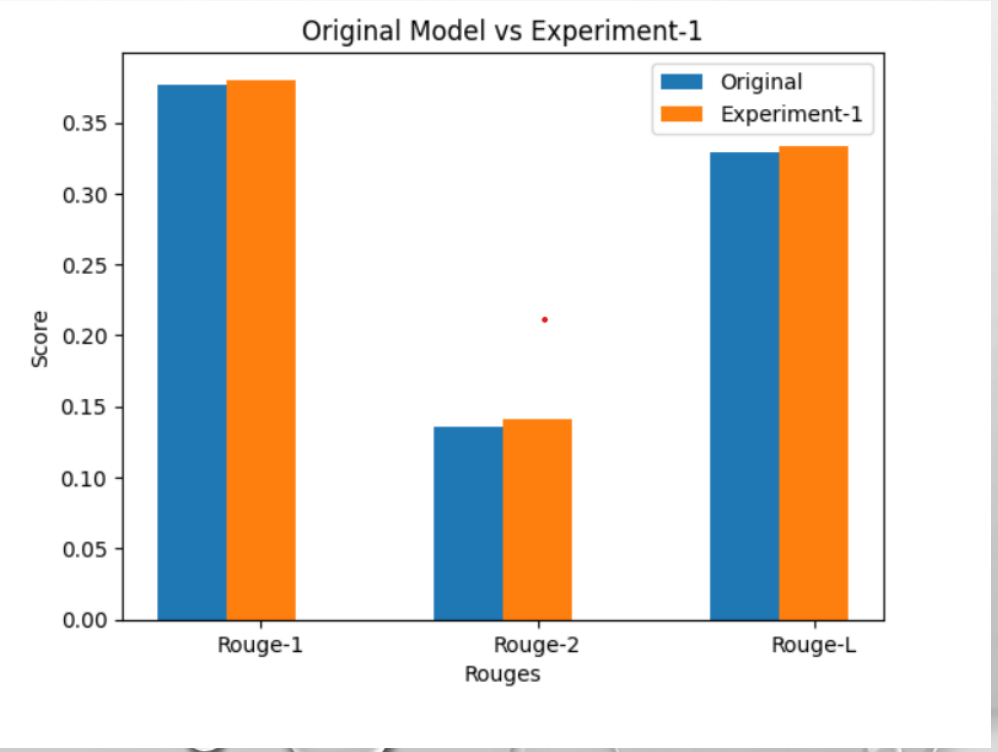
EXPERIMENTS CONDUCTED

- Experiment-1: Addition of a condition for checking edge between sentence nodes in the Sentence Graph step. The condition checked is **Co-Reference** between the pair of sentences. Co-reference is where two or more expressions refer to the same entity. A pronoun referring to a person or object in another sentence.
- Experiment-2: Using **Euclidean distance** to check sentence similarity instead of **Cosine Similarity** in Sentence Graph step.
- Experiment-3: Using alternate approaches for Extractive summarization.
- Experiment-4: Using seq2seq model for Abstractive Summarization
- Experiment-5: Using **pre-trained Hugging face Transformer** model for abstractive summarization.

RESULTS

- Experiment-1: The results of the experiment were better, having slightly better Rouge Scores than the original implementation.

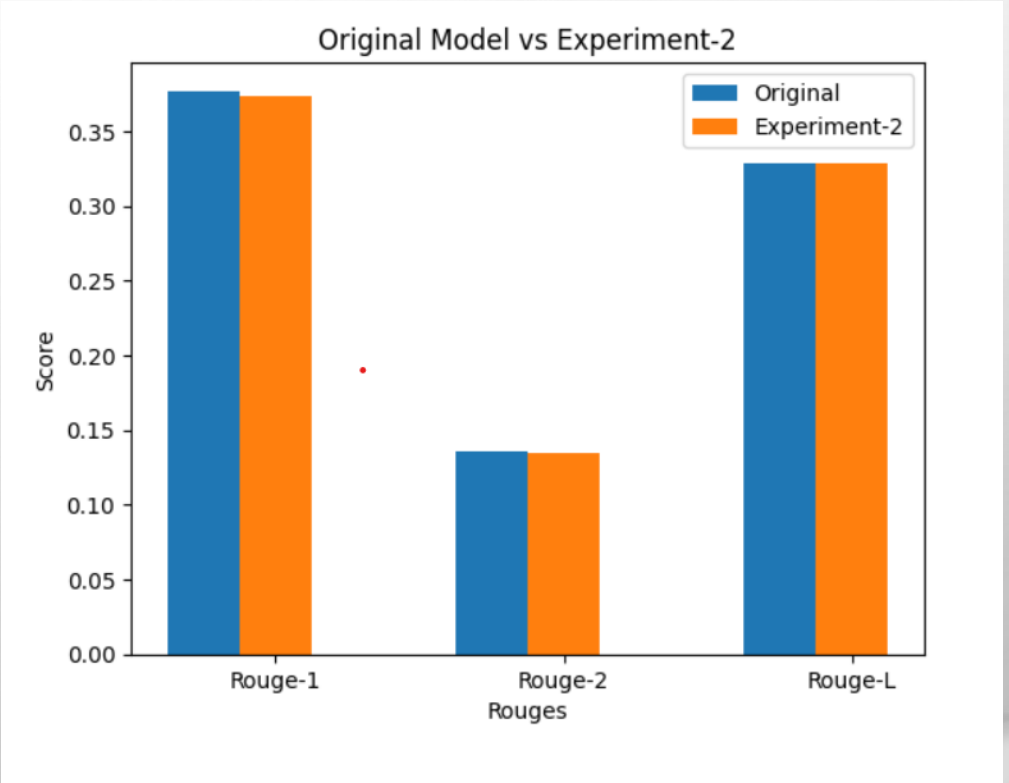
Model	Rouge-1	Rouge-2	Rouge-L
Original	0.377	0.136	0.329
Experiment-1	0.38	0.141	0.333



RESULTS

- Experiment-2: The results of the experiment were satisfactory having similar scores as the original implementation.

Model	Rouge-1	Rouge-2	Rouge-L
Original	0.377	0.136	0.329
Experiment-2	0.373	0.135	0.328



RESULTS

- Experiment 3: We have alternate approaches for extractive summarisation:
    1. Frequency-based approach
    2. Tf-Idf approach

```
ROGUE Scores for Frequency based approach
ROUGE-1: 0.2766996648238301
ROUGE-2: 0.08311613973407764
ROUGE-L: 0.24519889172116605
```

```
ROGUE Scores for Frequency based Tf-Idf approach
ROUGE-1: 0.29930268602339466
ROUGE-2: 0.10104034882506026
ROUGE-L: 0.26855892342746535
```

RESULTS

- Experiment 4: We have tried building seq2seq LSTM model for abstractive summarisation.
- The results were not very good as we couldn't train the model properly.
- We trained the model on the following dataset: (https://www.kaggle.com/datasets/news-summary)

	headlines	text
1542	'Fearless Girl' statue stares down Wall Street...	A statue of a young girl staring down the bron...
3659	Will look at Vodafone-Idea merger when require...	Telecom regulator TRAI on Friday said it will ...
2490	Indian fisherman shot dead by Sri Lankan Navy	An Indian fisherman was shot dead on Monday by...
3023	Salman, Aamir will run after 21-year-old heroi...	Raveena Tandon, when asked if she would like t...
3359	CBSE warns schools against selling books, unif...	Asserting that "educational institutions are n...

Model Results:

```
# 1 News:  man arrested delhi airport friday allegedly carrying nearly lakh unauthorised manner senior official said passenger identified rohit ku intercepted cisf personnel baggage check revealed suspicious packet containing five envelopes five envelopes contained
Original summary:  start man held cash delhi airport end
Predicted summary:   sostok bjp bjp govt govt start delhi end end end end end
ROUGE-1: 0.4615384565680473
ROUGE-2: 0.0
ROUGE-L: 0.4615384565680473
```

INDIVIDUAL CONTRIBUTION

- Yash Pathak – 2022201026

Implemented frequency-based, Tf-Idf for extractive summarization and seq2seq model for abstractive summarization.

- Gaurav Khapekar – 2022201055

Implemented and integrated spectral clustering and cluster compression components of SummPip.

- Vivek Kirpan – 2022201071

Implemented sentence graph component of SummPip and integration of entire SummPip with experiments. Also, played around with the pre-trained model of Hugging Face Transformer.