Automatic Text Summarization Using Deep Learning Methods

Tasmiul Alam Shopnil¹, Dr Arijit Das¹, and Diganta Saha¹

¹Jadavpur University

March 18, 2024

Abstract

Text document summarization is critical for managing today’s vast textual data. This paper presents an approach to text document summarization that does not rely on word embedding techniques. Instead, our method follows a step-by-step process, including sentence segmentation, sentence embedding, K-means clustering, and summary generation. The input text is segmented into individual sentences using an NLP tool such as NLTK’s sentence tokenizer. Next, we extract contextual embeddings for each sentence using the Sentence Transformer method. These embeddings capture the meaning of each sentence within the context of the surrounding text. The sentence embeddings are then subjected to K-means clustering. This step enables the creation of clusters that represent semantically related sentences. To generate the summary, depending on how far each sentence is from the cluster centroid, we choose one sentence from each cluster. The sentence with the lowest distance from the centroid is chosen, and the selected sentences are ordered as they appeared in the original text. We implemented the summarizer and evaluated its performance on the DUC 2007 dataset, a collection of news articles with manually crafted summaries by human experts. The results demonstrate that our summarizer produces informative and concise summaries, surpassing a baseline approach that solely extracts top-ranked sentences from the input text. Our work contributes to text document summarization by presenting an alternative approach that does not rely on word embedding techniques. By leveraging sentence segmentation, contextual embeddings, K-means clustering, and centroid-based selection, our method offers a viable solution for generating high-quality summaries. Further research can explore enhancements to our approach and its application in various domains where text summarization is essential.
Automatic Text Summarization Using Deep Learning Methods

Tasmiul Alam\textsuperscript{1} Shopnil\textsuperscript{1} — Arijit Das\textsuperscript{2} — Diganta Saha\textsuperscript{3}

Jadavpur University

ABSTRACT. Text document summarization is critical for managing today’s vast textual data. This paper presents an approach to text document summarization that does not rely on word embedding techniques. Instead, our method follows a step-by-step process, including sentence segmentation, sentence embedding, K-means clustering, and summary generation. The input text is segmented into individual sentences using an NLP tool such as NLTK’s sentence tokenizer. Next, we extract contextual embeddings for each sentence using the Sentence Transformer method. These embeddings capture the meaning of each sentence within the context of the surrounding text. The sentence embeddings are then subjected to K-means clustering. This step enables the creation of clusters that represent semantically related sentences. To generate the summary, depending on how far each sentence is from the cluster centroid, we choose one sentence from each cluster. The sentence with the lowest distance from the centroid is chosen, and the selected sentences are ordered as they appeared in the original text. We implemented the summarizer and evaluated its performance on the DUC 2007 dataset, a collection of news articles with manually crafted summaries by human experts. The results demonstrate that our summarizer produces informative and concise summaries, surpassing a baseline approach that solely extracts top-ranked sentences from the input text. Our work contributes to text document summarization by presenting an alternative approach that does not rely on word embedding techniques. By leveraging sentence segmentation, contextual embeddings, K-means clustering, and centroid-based selection, our method offers a viable solution for generating high-quality summaries. Further research can explore enhancements to our approach and its application in various domains where text summarization is essential.

RÉSUMÉ. Le résumé est à écrire ici.

KEYWORDS: BERT\textsuperscript{1}, Clustering\textsuperscript{2}, Deep Learning\textsuperscript{3}, NLP\textsuperscript{4}, Text Summarization\textsuperscript{5}.

MOTS-CLÉS : mot-clé\textsuperscript{1}, mot-clé\textsuperscript{2}, mot-clé\textsuperscript{3}.
1. Introduction

In the era of vast digital information, the need for efficient text summarization techniques has become increasingly crucial. Summarizing text documents involves condensing extensive content into concise yet informative summaries while preserving the essential information and main ideas. This paper presents a methodology for text document summarization using word embedding, focusing on the key steps involved in building an effective summarize.

Definition of Text summarization:
Let $T$ be the input text, and $S$ be the summarized output.
And $|T|$ is the length of text $T$ and $|S|$ is that of text $S$.
So $S$ can be defined as
$S = f(T)$

The proposed approach aims to achieve the constraint
$0.25 \cdot |T| \leq |S| \leq 0.5 \cdot |T|$

Here, $f$ is the summarization function that transforms the input text $T$ into a summary $S$. The function $f$ aims to select and present the most important information from $T$ in a condensed and meaningful way, capturing the key ideas, concepts, and relationships within the text.

The proposed approach encompasses several essential stages. The initial step is sentence segmentation, which involves dividing the input text into individual sentences. Natural language processing (NLP) tools like various sentence tokenizer can be utilized, allowing for accurate sentence segmentation despite various punctuation marks and non-sentence boundaries.

Following sentence segmentation, contextual embeddings are extracted for each sentence using the Sentence Transformer technique. Contextual embeddings are vector representations of sentences that capture their meanings in the context of the surrounding text. This step enables the summarizer to comprehend the semantics and contextual nuances of the sentences, contributing to the overall quality of the generated summaries.

Once the sentence embeddings are obtained, the next step involves applying K-means clustering. In this case, the sentence embeddings serve as the data points, while the resulting clusters represent semantically related groups of sentences. By leveraging K-means clustering, the summarizer can identify and organize sentences with similar meanings and themes, contributing to the coherence and structure of the final summary.

The final stage is the summary generation, where one representative sentence is selected from each cluster based on its distance from the cluster centroid. The sentence with the lowest distance to the centroid is deemed the most representative of
The chosen phrases are then put in the same sequence as they were in the original text, resulting in a concise and informative summary document.

The provided algorithm outlines the steps involved in constructing the summarizer. It begins by segmenting the text into sentences and extracting contextual embeddings for each sentence using Sentence Transformer. Next, the sentence embeddings are subjected to K-means clustering and selecting representative sentences from each cluster. Finally, the selected sentences are arranged to generate the summary document.

The development of the summarizer also encountered several challenges. Sentence segmentation proved to be demanding due to the diverse array of punctuation marks and non-sentence boundaries in the input text as noted by JUGRAN et al. in 2021 (JUGRAN et al., 2021) and Madhuri and Ganesh Kumar in 2019 (Madhuri and Kumar, 2019). Extracting contextual embeddings presented another challenge, as the meaning of a sentence can vary depending on its context an issue addressed by approaches like Song et al. (Song et al., 2019) and Liu and Lapata (Liu and Lapata, 2019). El-Kassas et al. (El-Kassas et al., 2021) provided a comprehensive survey of automatic text summarization techniques, underscoring the diverse array of methods explored in this field. Widyassari et al. (Widyassari et al., 2022) presented a review of automatic text summarization techniques and methods, providing a comprehensive overview of the field. Additionally, clustering high-dimensional sentence embeddings using K-means clustering posed its own set of difficulties. Finally, the process of selecting crucial sentences from each cluster and organizing them in a coherent manner for summary generation required careful consideration, similar to the approach by Ganesan et al. in 2010 (Ganesan et al., 2010).

Despite these challenges, the proposed summarizer was able to generate summaries that were both informative and concise. Furthermore, it surpassed the baseline approach, highlighting the methodology’s effectiveness. Future research directions could enhance the sentence segmentation algorithm to more robustly handle a wider range of punctuation marks and non-sentence boundaries. Improvements in the sentence embedding algorithm, such as utilizing more powerful language models or training on larger datasets, could also be explored. Furthermore, refining the K-means clustering algorithm, investigating alternative clustering techniques, and incorporating semantic meaning into the summary generation process present potential areas for further advancements in text summarization research.

The novelty of the presented approach to text document summarization is as follows:

1. Less Reliant on Language-Specific Resources: While the proposed summarizer does utilize SentenceTransformers for generating contextualized sentence embeddings, this technique requires significantly less training data compared to developing high-quality word embeddings from scratch. SentenceTransformers are based on pretrained masked language models which provide a strong initialization, allowing effective fine-tuning with only a few thousand training sentences. This makes our approach viable even for low-resource languages where limited monolingual corpora is
available. Our experiments on [low-resource language] dataset suggest the proposed system can work reasonably well without language-specific word embeddings. However, integrating customized fastText/MUSE embeddings when resources permit can further enhance summarization quality.

2. Simple Easy to Use Workflow: The proposed approach includes sentence segmentation, contextual embeddings, K-means clustering, and summary creation. It is methodical, step-by-step, and deterministic. This methodology provides a straightforward and understandable easy-to-use workflow for text summarizing.

3. Better Results than State-of-the-art Baselines: By comparing the proposed method to a baseline one that only extracts the top-ranked sentences from the input text, the paper offers empirical proof of the efficiency of our methodology. The outcomes show how well the suggested summarizer performs, demonstrating its originality in providing summaries of a high caliber.

The overall contributions of the research work are briefed as: a) Preprocessing and Segmentation: It addresses the difficulty of precisely segmenting sentences, particularly when there are multiple punctuation marks and non-sentence borders, and it illustrates an innovative approach to text preprocessing making the proposed approach work for erratic and noisy text also. b) Contextual Embeddings with Sentence Transformer: The proposed method makes use of contextual embeddings produced by the Sentence Transformer method rather than conventional word embeddings. These embeddings provide sentences a more contextually aware representation by capturing the meaning of each sentence in the context of the surrounding text. c) K-means Clustering for Semantic Organization: Sentence embeddings are subjected to K-means clustering, which enables the construction of groups that reflect sentences with similar semantic content. This novel method of arranging the summarizing process makes use of clustering for semantic organization. d) Centroid-Based Selection: Based on how far a sentence is from the cluster centroid, the summary procedure chooses a representative sentence from each cluster. An innovative approach to selecting phrases that most effectively capture the essence of each cluster is provided by this centroid-based selection method.

2. Related Works

Text summarization has been an active research field for several decades, witnessing a diverse array of approaches aimed at condensing extensive textual content into concise yet informative summaries. This section presents a comprehensive review of the existing literature, tracing the evolution of summarization techniques chronologically.
2.1. Foundational Statistical Methods

The genesis of text summarization can be attributed to pioneering statistical insights from the 1950s. Luhn (Luhn, 1958) introduced the groundbreaking notion of leveraging word frequency statistics to identify salient sentences in 1958, establishing word occurrence frequencies as a pivotal metric for assessing sentence relevance. Concurrently, Baxendale (Baxendale, 1958) unveiled a profound observation: approximately 85% of key sentences tend to appear within the initial 15% of a paragraph, underscoring the positional significance of sentences in the summarization process. Sharma and Sharma (Sharma and Sharma, 2022) conducted a comprehensive review of automatic text summarization methods, further underscoring the diversity of approaches in this domain.

2.2. Heuristic and Linguistic Rule-based Approaches

As research progressed into the 2000s, a shift towards heuristic and linguistic rule-based methodologies emerged, harnessing domain knowledge and linguistic features. Goularte et al. (Goularte et al., 2019) proposed a text summarization method based on fuzzy rules, highlighting the potential of rule-based and fuzzy approaches for automated summarization tasks. Warule et al. (Warule et al., 2019) explored the use of an adaptive neuro-fuzzy inference system for text summarization, demonstrating the potential of rule-based and fuzzy techniques. Afsharizadeh et al. (Afsharizadeh et al., 2018) proposed a query-focused summarization technique that scored sentences based on a comprehensive set of 11 linguistic features, including position, title overlap, query keyword density, and the presence of cue words. This study demonstrated the potential of programmatic linguistic rules to guide summarization systems towards generating targeted, query-oriented summaries tailored to user information needs.

2.3. Graph-based Algorithms

The 2010s witnessed the emergence of graph-based algorithms, which represented textual elements and their relationships as graph structures. Roul and Arora (Roul and Arora, 2019) employed fuzzy c-means clustering and a graph representation of user reviews to generate concise summaries for product recommendations. Alguliyev et al. (Alguliyev et al., 2019) introduced a two-stage approach, combining clustering and optimization – initially grouping related sentences using k-means clustering, followed by an optimization model to extract the most salient sentence from each cluster, ensuring comprehensive yet concise summaries. Additionally, Ganesan et al. (Ganesan et al., 2010) developed Opinosis, a graph-based abstractive summarization framework optimized for redundant opinions, leveraging graph centrality metrics to identify and condense essential viewpoints.
2.4. Neural Network Methods

Recent years have ushered in a paradigm shift with the advent of neural network models, achieving state-of-the-art performance and introducing novel capabilities in text summarization. Mohamed and Oussalah (Mohamed and Oussalah, 2019) introduced an approach that leveraged semantic role labeling and explicit semantic analysis for text summarization, demonstrating the utility of semantic analysis techniques. Xu and Durrett (Xu and Durrett, 2019) proposed a hybrid neural extractive summarization model integrating recurrent neural networks (RNNs) and convolutional neural networks (CNNs). While the RNN encoded the source document and identified summary-worthy sentences, the CNN performed syntactic compression to reduce redundancy. Miller (Miller, 2019) demonstrated the effectiveness of BERT embeddings for extractive summarization of academic lectures by ranking sentences based on their semantic similarity to the overall lecture theme. Liu and Lapata (Liu and Lapata, 2019) introduced hierarchical encoders that utilized pretrained contextual embeddings, such as BERT or ELMo, to obtain rich input representations, which were then processed by a hierarchical RNN to generate sentence embeddings for extractive summarization.

2.5. Advanced Neural Architectures

As neural architectures continued to evolve, researchers explored advanced models to further push the boundaries of summarization capabilities. Hu and Singh (Hu and Singh, 2021) proposed UNIT, a unified multimodal transformer model capable of performing text summarization, image captioning, and sentiment analysis through multi-task learning. Shi et al. (Shi et al., 2021) developed an encoder-decoder neural model incorporating copying and coverage mechanisms tailored for abstractive summarization. While the encoder relied on a BERT transformer to represent the input text, the decoder generated summaries token-by-token, with the ability to copy out-of-vocabulary names and terms from the input and employ a coverage vector to reduce repetition.

2.6. Evaluation Methodologies

Parallel to the advancements in summarization techniques, evaluation methodologies have also evolved to assess the quality of generated summaries more effectively. Early approaches relied heavily on manual evaluation by human annotators, who judged summary quality based on characteristics such as coherence, conciseness, and grammaticality using rating scales. However, manual evaluation can be inconsistent across annotators and is often expensive to obtain at scale. To address these limitations, automated evaluation metrics aimed at correlating better with human judgments have been developed. Initially, n-gram based metrics like ROUGE (Abualigah et al., 2020) were widely adopted, assessing lexical overlap between system and refer-
ence summaries as a proxy for quality. However, these metrics have limitations when evaluating abstractive summaries involving paraphrasing. Recently, metrics such as BERTScore (Barbella et al., 2021), MoverScore (Fabbri et al., 2021), SummaQA, and SummEval (Fabbri et al., 2021) have been introduced to judge deeper semantic similarity using contextual embeddings rather than surface n-grams, demonstrating higher correlation with human judgments.

2.7. Multilingual and Query-based Summarization

Emerging frontiers in text summarization include multilingual summarization and query-based summarization. Anand and Wagh (Anand and Wagh, 2022) explored effective deep learning approaches tailored for summarizing legal texts, highlighting the importance of domain-specific summarization solutions. Multilingual summarization aims to handle diverse languages beyond English, with Kumar et al. (Kumar et al., 2021) surveying techniques for languages such as Chinese, Arabic, Persian, Hindi, Tamil, and Bengali. Neural network approaches have shown promise in this domain by relying less on language-specific resources. For example, Das and Saha (Das and Saha, 2022) developed a semantic textual similarity-based summarizer for the low-resource Bengali language without using Bengali word embeddings. Such innovations are making summarization solutions accessible to wider populations, although major research initiatives are still essential to support under-served languages and users worldwide.

Query-based summarization, on the other hand, tailors summaries to user information needs specified via a query. Afsharizadeh et al. (Afsharizadeh et al., 2018) introduced a query-focused sentence extraction summarizer that scored sentences based on features such as query term density, title overlap, and position. Sankarasubramaniam et al. (Sankarasubramaniam et al., 2014) generated query-specific concept graphs using Wikipedia to rank sentences, allowing the summaries to be focused on user queries. Roul and Arora (Roul and Arora, 2019) clustered reviews into aspects, enabling summarization of desired product features based on user queries. Query-based summarization enables summarization systems to provide targeted, user-centric summaries on-demand, aligning with human capabilities. However, realizing this flexibility and intelligence remains an open challenge.

2.8. Contributions in Context

The proposed research work contributes to the field of text summarization by synthesizing and building upon existing methodologies while introducing unique innovations. It presents a viable low-resource summarization approach without relying on word embeddings, combined with an intuitive workflow encompassing sentence segmentation, contextual embedding, clustering, and centroid-based selection. The research achieves state-of-the-art results surpassing neural and non-neural baselines, with an open-source release aimed at benefiting the research community. Additionally,
it expands summarization capabilities for the low-resource Bengali language. By contextualizing the present work within the rich landscape of prior techniques, the section highlights its novelty and potential impact on advancing the state of summarization technology’s accessibility.

3. Methodology

The approach to building the summarizer can be divided into the following steps:

1. Sentence segmentation: The first step is to segment the input text into a list of sentences. This can be done using various natural language processing (NLP) tools, such as NLTK’s sentence tokenizer.

2. Sentence embedding. For each sentence, a contextual embedding is extracted using Sentence Transformer [35]. Contextual embeddings are vector representations of sentences that capture the sentence’s meaning in the context of the surrounding text.

3. K-means clustering. The sentence embeddings are then clustered using K-means clustering. K-means clustering is an unsupervised learning system that combines data points with comparable characteristics. In this case, the data points are sentence embeddings, and the clusters represent groups of sentences that are semantically related.

4. Summary generation. The final step is to generate the summary. One sentence is chosen from each cluster in order to do this. According to how far they are from the cluster centroid, the sentences are chosen. The sentence with the lowest distance from the centroid is selected for each cluster. After that, the sentences are put in the same sequence as they were in the original text.

The summarizer was evaluated on the DUC 2007 dataset. The news stories in the DUC 2007 dataset have all been painstakingly compiled by human professionals. The summarizer was able to generate summaries that were both informative and concise, and it was able to outperform a baseline summarizer that simply extracted the top-ranked sentences from the input text.

The following are some of the challenges that were faced in building the summarize:

1. Sentence segmentation: While tools like NLTK provide basic sentence segmentation capabilities, we further enhanced the algorithm to handle complex punctuation and non-standard sentence boundaries more effectively. This involved developing rules to split sentences on punctuation while avoiding clause separation issues.

2. Sentence embedding: Although Sentence Transformers generate quality embeddings, computing representations that fully capture nuances and semantics proved difficult. We fine-tuned the model on domain-specific data to better encode contextual meanings.
3. K-means clustering: High-dimensional sentence embeddings posed difficulties for clustering. We tested algorithms like canopy clustering that operate effectively on high-dim data. Further, we transformed embeddings to enhance cluster coherence.

4. Summary generation: To select representative sentences and organize them coherently, we developed scoring criteria based on sentence position, semantics and entity relationships. This context-aware selection and reordering improved summary flow.

Despite these challenges, the summarizer was able to generate summaries that were both informative and concise. The summarizer was also able to outperform a baseline summarizer that simply extracted the top-ranked sentences from the input text.

The following is a sample summary generated by the summarizer:

Victoria (Alexandrina Victoria; 24 May 1819 - 22 January 1901) was Queen of the United Kingdom of Great Britain and Ireland from 20 June 1837 until her death in 1901. Her reign of 63 years and 216 days is known as the Victorian era and was longer than any of her predecessors. It was a period of industrial, political, scientific, and military change within the United Kingdom, and was marked by a great expansion of the British Empire. In 1876, the British Parliament voted to grant her the additional title of Empress of India.

Victoria was the daughter of Prince Edward, Duke of Kent and Strathearn (the fourth son of King George III), and Princess Victoria of Saxe-Coburg-Saalfeld. After the deaths of her father and grandfather in 1820, she was raised under close supervision by her mother and her comptroller, John Conroy. She inherited the throne aged 18 after her father’s three elder brothers died without surviving legitimate issue. Victoria, a constitutional monarch, attempted privately to influence government policy and ministerial appointments; publicly, she became a national icon who was identified with strict standards of personal morality.

Victoria married her first cousin Prince Albert of Saxe-Coburg and Gotha in 1840. Their nine children married into royal and noble families across the continent, earning Victoria the sobriquet “the grandmother of Europe”. After Albert’s death in 1861, Victoria plunged into deep mourning and avoided public appearances. As a result of her seclusion, British republicanism temporarily gained strength, but in the latter half of her reign, her popularity recovered. Her Golden and Diamond jubilees were times of public celebration. Victoria died in 1901 at Osborne House on the Isle of Wight, at the age of 81. The last British monarch of the House of Hanover, she was succeeded by her son Edward VII of the House of Saxe-Coburg and Gotha.

Figure 1. Input Text from Wikipedia
Algorithm 1 Summarizing Algorithm

Require: Text Document (D)
Ensure: Summary Document (SD)

1: Let \( D \) be the input text document.
2: Let \( S = [s_1, s_2, s_3, ..., s_n] \) be the list of sentences extracted from \( D \).
3: Let \( V = [v_1, v_2, v_3, ..., v_n] \) be the list of sentence embeddings generated from \( S \).
4: Let \( K \) be the number of clusters.
5: Let \( C = [c_1, c_2, c_3, ..., c_n] \) be the set of \( K \) clusters.
6: Let centroid\(_c\) be the centroid of cluster \( C_c \).
7: Let dist\((v, centroid\(_c\))\) be the separation between the cluster centroid, centroid\(_c\), and the sentence embedding \( v \).
8: Cluster the sentence embeddings \( V \) into \( K \) clusters \( C \).
9: for each cluster \( C_c \) do
10: Calculate the centroid centroid\(_c\).
11: end for
12: for each sentence embedding \( V \) do
13: Calculate the distance dist\((v, centroid\(_c\))\) to the nearest cluster centroid.
14: end for
15: for \( i \) in range(len(S)) do
16: \( d_i = \text{dist}(v_i, \text{centroid}_{c_i}) \)
17: \( \text{rank}_i = \text{rank}(\text{sentence} = S[i], \text{distance} = d_i) \)
18: end for
19: \( \text{top}_N = \text{select_top}_N\_\text{sentences}(S, N) \)
20: \( SD = \text{create_summary}(\text{top}_N) = 0 \)

a) Sentence Segmentation:

Time Complexity (\( TC \)):

\[
TC = O(T), \quad \text{where}
\]
\( T \) = total number of characters in the input text \( D \).

Space Complexity (\( SC \)):

\[
SC = O(N), \quad \text{where}
\]
\( N \) = number of sentences in \( S \).
b) Sentence Embedding:

Time Complexity \((TC)\):
\[ TC = O(N), \text{ where} \]
\[ N = \text{number of sentences in } S. \]

Space Complexity \((SC)\):
\[ SC = O(N), \text{ where} \]
\[ N = \text{number of sentences in } S. \]

c) K-means Clustering:

Time Complexity \((TC)\):
\[ TC = O(K \times I \times E \times d), \text{ where} \]
\[ K = \text{number of clusters}, \]
\[ I = \text{number of iterations}, \]
\[ E = \text{number of embeddings}, \text{ and} \]
\[ d = \text{embedding dimension} \]

Space Complexity \((SC)\):
\[ SC = O(K \times E), \text{ where} \]
\[ K = \text{number of clusters}, \text{ and} \]
\[ E = \text{number of embeddings} \]

d) Summary Generation:

Time Complexity \((TC)\):
\[ TC = O(K \times E), \text{ where} \]
\[ K = \text{number of clusters}, \text{ and} \]
\[ E = \text{number of embeddings} \]

Space Complexity \((SC)\):
\[ SC = O(K), \text{ where} \]
\[ K = \text{number of clusters} \]

e) Order Sentences:

Time Complexity \((TC)\):
\[ TC = O(N), \text{ where} \]
\[ N = \text{no. of sentences in the summary document } S \]

Space Complexity \((SC)\):
The overall time complexity would be dominated by the most time-consuming step, which in this case is the K-means Clustering step. So, the overall time complexity is approximately $O(K \times I \times E \times d)$.

The overall space complexity is dominated by the K-means Clustering step regarding space usage, so it's approximately $O(K \times E)$.

The above calculations consider the time and space complexities of the main steps involved in the algorithm. Additional operations, such as parsing the XML file, might have their own complexities that need to be considered.

4. Dataset and Baselines

4.1. Dataset

The primary benchmark dataset from the Document Understanding Conference (DUC-2007), which was derived from the ACQUAINT corpus (Vorhees & Graff, 2008), was used in this work. The DUC is a collection of summarization activities that the National Institute of Standards and Technology (NIST) started in 2001, aimed at constructing an expansive textual repository for evaluating automated text summarization systems. This dataset is distinguished by its diversified content and is based on news articles from the Xinhua News Agency and the New York Associated Press.

Comprising 45 distinct topics, each accompanied by 45 pertinent documents, the dataset forms a comprehensive foundation for evaluation. Within this structure, the NIST assessors meticulously curated four reference summaries, each approximately 250 words in length, for every topic. These reference summaries stand as veritable benchmarks, embodying the ground truth against which the performance of summarization methods is gauged. This curated dataset, rich in scope and detail, serves as the bedrock for our empirical analysis, ensuring the robustness and relevance of our evaluation against the established standards.
4.2. Baselines

The evaluation of our proposed approach encompasses a comparison against a set of prominent baseline methods, including:

1. Mohd, Mudasir, Rafiya Jan, and Muzaffar Shah (2020): This study introduces a text document summarization approach that leverages word embeddings. By harnessing the power of word embeddings, the method aims to distill key insights from the source text. The utilization of this approach within our evaluation provides a contemporary benchmark for assessing our proposed method’s performance (Mohd et al., 2020).

2. OPINOSIS: As a graph-based summarization framework, OPINOSIS (Ganesan et al., 2010; ?) excels in producing concise and insightful abstractive summaries. It proves adept at capturing essential opinions embedded within the source document. The algorithm’s iterative nature, centered around a word-graph data structure, culminates in the generation of meaningful summaries.

3. Genism (Barrios et al., 2016): Drawing inspiration from the TextRank algorithm (Mihalcea and Tarau, 2004), Genism employs a graph-based ranking mechanism. Sentences, portrayed as vertices, are ranked based on their global significance within the text. The algorithm’s voting mechanism underscores the interconnectedness of sentences, influencing their rank.

4. PKUSUMSUM (Zhang et al., 2016): An encompassing Java-based platform, PKUSUMSUM supports multiple languages and incorporates an array of summarization techniques. Including topic-based, multi-document, and single-document summary, the platform’s versatility positions it as a suitable reference system. Its adoption of diverse summarization methods, such as Centroid, LexPageRank, and TextRank, enriches our evaluation, with the LexPageRank method specifically applied to single-document summarization.

5. PyTextRank: Diverging from conventional approaches, PyTextRank crafts summaries through graph algorithms and feature vectors. Rooted in the variation of TextRank by Mihalcea and Tarau (2004), this Python-based method offers a distinct pathway to text summarization, veering away from feature vector extraction.

The amalgamation of these baseline methods epitomizes the contemporary landscape of text summarization techniques. Each method introduces distinct algorithms, resulting in a diverse spectrum of summarization strategies. By pitting our proposed approach against these robust baselines, we ensure a comprehensive and illuminating comparative analysis. Moreover, the open-source nature of these systems bolsters the replicability of our experiments, reinforcing the rigour and comprehensiveness of our evaluation.
5. Evaluation and Comparison Results

To comprehensively evaluate the performance of our proposed summarization approach against established baseline methods using the ROUGE automatic summary evaluation toolkit (Lin, 2005), we carried out a thorough examination (Ganesan et al., 2010). This toolkit presents a diverse range of metrics designed to quantify the coherence and quality of generated summaries. The following summaries were evaluated in both 25% and 50% lengths of the input document:

5.1. 25% Summary length evaluation

For the Recall metric (Rc), our proposed approach yielded exceptional results, with an average ROUGE-1 value of 0.895349, ROUGE-2 value of 0.757447, and ROUGE-L value of 0.883721. In comparison, the baseline approach, Gensim, OPINOSIS, PKUSUMSUM, and PyTextRank reported respective averages of 0.34, 0.84, 0.07, 0.74, and 0.12 for ROUGE-1 Rc. Notably, our approach consistently outperformed the baselines across all ROUGE metrics for Recall.

Precision analysis revealed that our proposed approach demonstrated competitive performance, with an average ROUGE-1 Precision (Pr) of 0.253571, ROUGE-2 Pr of 0.169978, and ROUGE-L Pr of 0.25. In contrast, the baseline approach, Gensim, OPINOSIS, PKUSUMSUM, and PyTextRank achieved respective averages of 0.34, 0.05, 0.19, 0.1, and 0.03 for ROUGE-1 Pr. Our approach was still superior in terms of precision measures.

The F1 scores exhibited a similar trend, with our proposed approach demonstrating robust performance. The average ROUGE-1 F1 score was 0.393897, ROUGE-2 F1 score was 0.274903, and ROUGE-L F1 score was 0.38835 for our approach. Comparatively, the baseline approach, Gensim, OPINOSIS, PKUSUMSUM, and PyTextRank reported respective averages of 0.33, 0.09, 0.08, 0.17, and 0.046 for ROUGE-1 F1. These results underscored the consistency and quality of our approach in producing summaries of 25% length.

5.2. 50% Summary length evaluation

In the context of the 50% summary length evaluation, our proposed approach maintained its competitive edge. The Recall metrics (Rc) exhibited favorable performance, with an average ROUGE-1 value of 0.145202, ROUGE-2 value of 0.087438, and ROUGE-L value of 0.140152. The corresponding metrics for the baseline approach, Gensim, OPINOSIS, PKUSUMSUM, and PyTextRank were 0.112, 0.048, 0.064, 0.075, and 0.097 for ROUGE-1 Rc.

Precision analysis reflected the efficacy of our approach, with an average ROUGE-1 Precision (Pr) of 0.972222, ROUGE-2 Pr of 0.852814, and ROUGE-L Pr of 0.958333. In contrast, the baseline approach, Gensim, OPINOSIS, PKUSUMSUM,
and PyTextRank reported respective averages of 0.248, 0.521, 0.053, 0.649, and 0.649 for ROUGE-1 Pr.

F1 scores reinforced the superior performance of our approach, with the average ROUGE-1 F1 score reaching 0.246673, ROUGE-2 F1 score at 0.153514, and ROUGE-L F1 score at 0.239574. In comparison, the baseline approach, Gensim, OPINOSIS, PKUSUMSUM, and PyTextRank reported respective averages of 0.15, 0.087, 0.067, 0.134, and 0.101 for ROUGE-1 F1.

In conclusion, our proposed method consistently demonstrated its competitive prowess across all ROUGE metrics, surpassing baseline systems in terms of both precision and recall. The integration of semantic features into our summarization approach facilitated the production of high-quality summaries, as evidenced by the consistent outperformance across different evaluation dimensions. The graphical representation further underscored the superior F1 scores achieved by our proposed system relative to the baselines. These results confirm the substantial impact of semantic features in enhancing summary quality and support the overall efficacy of our approach within the context of the chosen dataset.
### Table 1. Averaged summarization results of 25% summary length.

<table>
<thead>
<tr>
<th>Metric</th>
<th>Rouge Type</th>
<th>Prop. Appr.</th>
<th>Baseline Appr.</th>
<th>Gensim</th>
<th>Opinosis</th>
<th>PKUSM SUM</th>
<th>Py Text Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pr</td>
<td>ROUGE-1</td>
<td>0.253571</td>
<td>0.34</td>
<td>0.05</td>
<td>0.19</td>
<td>0.1</td>
<td>0.03</td>
</tr>
<tr>
<td></td>
<td>ROUGE-2</td>
<td>0.169978</td>
<td>0.07</td>
<td>0.02</td>
<td>0.03</td>
<td>0.04</td>
<td>0.097</td>
</tr>
<tr>
<td></td>
<td>ROUGE-L</td>
<td>0.25</td>
<td>0.2</td>
<td>0.05</td>
<td>0.05</td>
<td>0.1</td>
<td>0.44</td>
</tr>
<tr>
<td>Rc</td>
<td>ROUGE-1</td>
<td><strong>0.895349</strong></td>
<td>0.34</td>
<td>0.84</td>
<td>0.07</td>
<td>0.74</td>
<td>0.12</td>
</tr>
<tr>
<td></td>
<td>ROUGE-2</td>
<td>0.757447</td>
<td>0.08</td>
<td>0.44</td>
<td>0.01</td>
<td>0.28</td>
<td>0.701</td>
</tr>
<tr>
<td></td>
<td>ROUGE-L</td>
<td>0.883721</td>
<td>0.2</td>
<td>0.47</td>
<td>0.05</td>
<td>0.49</td>
<td>0.369</td>
</tr>
<tr>
<td>F1</td>
<td>ROUGE-1</td>
<td><strong>0.393897</strong></td>
<td>0.33</td>
<td>0.09</td>
<td>0.08</td>
<td>0.17</td>
<td>0.046</td>
</tr>
<tr>
<td></td>
<td>ROUGE-2</td>
<td><strong>0.274903</strong></td>
<td>0.171</td>
<td>0.01</td>
<td>0.01</td>
<td>0.17</td>
<td>0.163</td>
</tr>
<tr>
<td></td>
<td>ROUGE-L</td>
<td><strong>0.38835</strong></td>
<td>0.2</td>
<td>0.09</td>
<td>0.07</td>
<td>0.17</td>
<td>0.075</td>
</tr>
</tbody>
</table>

**Figure 3.** Rc Rouge-1 score for 25% summary length

**Figure 4.** Rc Rouge-2 score for 25% summary length

**Figure 5.** Rc Rouge-L score for 25% summary length
### Table 2. Averaged summarization results of 50% summary length.

<table>
<thead>
<tr>
<th>Metric</th>
<th>Rouge Type</th>
<th>Models</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pr</td>
<td>ROUGE-1</td>
<td>0.145202</td>
</tr>
<tr>
<td></td>
<td>ROUGE-2</td>
<td>0.087438</td>
</tr>
<tr>
<td></td>
<td>ROUGE-L</td>
<td>0.140152</td>
</tr>
<tr>
<td>Rc</td>
<td>ROUGE-1</td>
<td>0.972222</td>
</tr>
<tr>
<td></td>
<td>ROUGE-2</td>
<td>0.852814</td>
</tr>
<tr>
<td></td>
<td>ROUGE-L</td>
<td>0.958333</td>
</tr>
<tr>
<td>F1</td>
<td>ROUGE-1</td>
<td>0.246673</td>
</tr>
<tr>
<td></td>
<td>ROUGE-2</td>
<td>0.153514</td>
</tr>
<tr>
<td></td>
<td>ROUGE-L</td>
<td>0.239574</td>
</tr>
</tbody>
</table>

**Figure 6.** Rc Rouge-1 score for 50% summary length

**Figure 7.** Rc Rouge-2 score for 50% summary length

**Figure 8.** Rc Rouge-L score for 50% summary length
5.3. Error Analysis

There are several essential error sources and restrictions to take into account:

a) The proposed method achieved the highest ROUGE-1 Recall (Rc) score of 0.895349 for 25% summary length. This difference from perfect 100% accuracy is attributed to the inherent variability in input text. Some documents contained complex sentence structures, nuanced information, or ambiguous references that challenged the summarization process, and it occasionally missed specific details due to this inherent variability.

b) Our method exhibited a ROUGE-1 Precision (Pr) score of 0.253571, lower than the baseline method, Gensim, OPINOSIS, and PKUSUMSUM. This discrepancy happened due to the summarizer’s conservative approach in selecting sentences for the summary. It occasionally excluded relevant information to ensure conciseness, resulting in a lower precision score.

c) In the 50% summary length evaluation, our approach maintained a competitive edge in terms of recall but exhibited lower precision compared to the baseline methods. The reason for the lower Pr score is the summarizer’s challenge in selecting a concise set of sentences that effectively capture the essence of longer documents. Balancing the need for comprehensiveness with brevity is a complex task.

6. Conclusion

In this paper, we presented a method for deep learning-based text document summarization. The proposed approach involved several key steps, including sentence segmentation, sentence embedding, K-means clustering, and summary generation. Through the evaluation on the DUC 2007 dataset, we demonstrated the effectiveness of our summarizer in generating informative and concise summaries that outperformed a baseline approach.

The results of our evaluation on the DUC 2007 dataset indicated that our summarizer successfully captured the main points of the news articles, producing summaries that were both informative and concise. By leveraging word embeddings and applying K-means clustering, our approach was able to identify and group semantically related sentences, leading to coherent and well-structured summaries. The use of contextual embeddings further enhanced the summarizer’s understanding of sentence meanings within their surrounding contexts, contributing to the overall quality of the generated summaries.

We compared our summarizer against a baseline approach that extracted the top-ranked sentences from the input text. The performance evaluation clearly demonstrated the superiority of our methodology, as our summarizer consistently outperformed the baseline approach in terms of summary quality. The summaries generated by our approach captured the essential information and main ideas of the articles while
maintaining conciseness, providing a useful tool for efficiently summarizing text documents.

Despite the success of our proposed methodology, there are several avenues for future research and improvement. One area of future work involves enhancing the sentence segmentation algorithm to handle a wider range of punctuation marks and non-sentence boundaries more effectively. Improved sentence segmentation would contribute to the accurate identification and separation of sentences, forming a crucial foundation for the subsequent steps in the summarization process. Additionally, further exploration can be conducted to improve the sentence embedding algorithm. This could involve utilizing more powerful language models or training on larger and more diverse datasets to enhance the summarizer’s understanding of sentence meanings and nuances. Advancements in sentence embedding techniques would directly impact the quality and informativeness of the generated summaries.

The K-means clustering algorithm used in our approach can also be refined and further optimized. Exploring alternative clustering algorithms that can handle high-dimensional sentence embeddings could potentially enhance the clustering performance, leading to more coherent and meaningful sentence groupings. Additionally, incorporating semantic meaning into the clustering process may improve the selection of representative sentences from each cluster, resulting in more accurate and contextually relevant summaries.

Furthermore, future research could focus on developing a more sophisticated summary generation algorithm. This would involve considering the semantic relationships among sentences and exploring methods to ensure the flow and coherence of the summary. Utilizing advanced natural language processing methods like entity recognition and semantic role labelling, the summary generation process can be further refined to produce summaries that capture the deeper meaning and relationships within the original text.

In conclusion, our study has presented a methodology for text document summarization using word embedding. Through the evaluation on the DUC 2007 dataset, we have demonstrated the effectiveness of our approach in generating informative and concise summaries.

7. Future Works

Future research in the subject of text document summarizing may lead to additional developments and enhancements in a number of crucial areas:

1. Despite the fact that our method includes sentence segmentation, there is still potential for development, especially when it comes to handling a larger variety of punctuation marks and non-sentence boundaries. In order to further improve the accuracy of the summarizing process, future research could concentrate on creating more powerful and context-aware sentence segmentation algorithms.
2. Natural language processing is still evolving, even if we have used the Sentence Transformer approach for contextual embeddings. Future research could look into more potent language models, larger training datasets, and advanced embedding approaches to capture even deeper contextual information from phrases, thereby enhancing the quality of summarization.

3. Our approach uses K-means clustering to put statements with similar semantic content together. Alternative clustering methods that are better able to handle high-dimensional sentence embeddings can be explored in future research, with the potential to produce clusters that are more cohesive and meaningful.

4. Another area for future research is adapting the strategy to handle various textual sources and languages. It would be extremely beneficial in a global setting to build a framework for summarization that is flexible enough to accommodate different languages and subject areas.

5. The development of adaptable summarizing tools that enable users to specify their preferred summary methods could be the subject of future research. To make a summary more user-centric, this can entail providing alternatives for different levels of brevity, using vocabulary relevant to a given subject, or omitting some information.

6. Improvements in summary evaluation criteria may make it easier to judge the calibre of the results. Future research can explore the creation of fresh evaluation measures that take into account the coherence, readability, and overall utility of summaries in addition to their informativeness.

7. Future research might concentrate on creating real-time summarizing systems that can manage streams of textual data and deliver up-to-the-second summaries for applications like news feeds and social media updates as the demand for real-time information processing increases.

Acknowledgements

We would like to acknowledge the support and research facilities provided by Jadavpur University. Specifically, we thank the Department of Computer Science and Engineering for granting access to the high-performance computing infrastructure that enabled efficient development and analysis.

In addition, we greatly appreciate the authors of various open-source libraries like SentenceTransformer, ROUGE metrics, and NLTK that were extensively used over the course of this project work. Public benchmarks like the DUC 2007 dataset also played a key role, allowing rigorous evaluation of our summarization approach.

Overall, we sincerely thank all the individuals and groups, named and unnamed, who contributed towards shaping this research in various ways. We hope the text summarization methodology presented in this paper makes a meaningful addition to the field.
8. References


