Scalable Computing: Practice and Experience, ISSN 1895-1767, http://www.scpe.org

Volume 25, Issues 2, pp. 972–986, DOI 10.12694/scpe.v25i2.2556

TEXT SUMMARIZATION FOR ONLINE AND BLENDED LEARNING

MAHIRA KIRMANI* AND GAGANDEEP KAUR†

Abstract. Online learning text summarization is vital for managing the constant influx of online information. It involves condensing lengthy online content into concise summaries while retaining the original meaning and information. While several online summarization tools are available, they often fall short in preserving the underlying semantics of the text. In this paper, we introduce an innovative approach to online text summarization that strongly emphasizes capturing and preserving the semantics of the text. Our automatic summarizer leverages distributional semantic models to extract and incorporate semantics, producing high-quality online summaries. To evaluate the effectiveness of our online summarization system, we conducted experiments on a diverse range of online content. We employed ROUGE metrics, a popular evaluation method for text summarization, to assess our system's performance. Additionally, we compared our results with those of four state-of-the-art online summarizers. The outcome of our study demonstrates that our online summarization approach, which integrates semantics as a fundamental feature, outperforms other reference summarizers. This conclusion underscores the significance of leveraging semantics in the context of online learning text summarization. Furthermore, our system's ability to reduce redundancies in online content makes it a valuable tool for managing information overload in the digital age.

Key words: Text Summarization, Online Learning, Semantic Models, Digital Education, Semantic-enhanced Summarization

AMS subject classification. 6804

1. Introduction. In recent years, the proliferation of online and blended learning environments has revolutionized education, offering unprecedented flexibility and accessibility to learners worldwide. With the influx of digital content, ranging from educational materials to scholarly articles, the need for efficient information processing has become more pronounced than ever before. Text summarization, a transformative natural language processing (NLP) technique, emerges as a crucial solution to address the challenges posed by information overload in the context of modern learning paradigms. Text summarization involves the condensation of textual information while retaining its essential meaning and context. This process holds immense promise in enhancing the learning experience for both educators and students [21]. Educators can leverage summarization techniques to distil voluminous content into concise and digestible formats, facilitating better knowledge dissemination and promoting effective communication. Learners, on the other hand, can benefit from the succinct summaries that aid comprehension, retention, and revision.

The application of text summarization techniques in the realm of education introduces a multifaceted landscape of opportunities and challenges. Adaptation of these techniques to the unique characteristics of online and blended learning environments necessitates an in-depth exploration of the intersection between NLP, education, and technology. Factors such as the diversity of content types, varying levels of technical proficiency among learners, and the need for personalized learning experiences demand a nuanced approach to text summarization in this domain.

This paper embarks on a comprehensive journey to delve into the realm of text summarization tailored specifically for online and blended learning environments. We aim to investigate state-of-the-art methodologies, explore their potential applications, and evaluate their impact on enhancing educational practices. By scrutinizing the challenges that arise when applying these techniques to diverse educational contexts, we aspire to provide valuable insights into the design and implementation of effective text summarization systems.

*University Institute of Computing, Chandigarh University, NH-05-Chandigarh-Ludhiana, Mohali, Punjab, India (mahirakirmani68@gmail.com)

†University Institute of Computing, Chandigarh University, NH-05-Chandigarh-Ludhiana, Mohali, Punjab, India (gagandeepkaurlogani@gmail.com)

Through an amalgamation of NLP advancements, pedagogical insights, and technological innovations, we strive to pave the way for a more streamlined and efficient transfer of knowledge in online and blended learning settings. As we traverse the intricacies of text summarization's integration into education, we anticipate that this study will contribute to the broader discourse on leveraging NLP techniques to reshape the landscape of modern learning, making it more engaging, accessible, and impactful for learners across the globe [42].

We aim to design a semantic text summarizer capable of summarizing English language learning material for online learning. Our summarizer uses distributional semantic models for summarization [34]. Through this, we endeavour to design a system that bridges the gap between semantic understanding and effective content summarization, ultimately contributing to a more efficient and impactful transfer of knowledge within digital learning ecosystems [15]. However, this fusion of semantic models and summarization is not without its challenges [36]. Adapting semantic understanding to the diverse subject matter, writing styles, and educational levels present in digital content requires careful consideration [35]. The intricate interplay between semantics and summarization mechanics demands sophisticated algorithmic approaches and robust model training.

Our proposed summarizer for online and blended learning text consists of the following steps:

(1) We use semantics as a feature for obtaining summaries of the learning material. The learning material is written in the English language, and we use general-purpose English language distributional semantic models to obtain semantics. These models are domain-independent and do not require any linguistic training. We use these models in a customized and novel way that transforms each sentence in the learning material into its semantic mappings. We call these semantic mappings as big-vectors. The big-vectors are thus our way of transforming text in the learning material into its corresponding semantic extension. Precisely, every sentence from the learning material is tokenized into words. These words are then fed to th semantic model to obtain a word vector for that particular word. The word vectors obtained in an individual sentence of text are concatenated to obtain a unified vector, which we call big-vector. Thus, we have its corresponding big-vector for each sentence in the learning material.

2) Next, we apply the k-means clustering algorithm to the big-vectors to obtain k semantic clusters.

3) We then apply our novel ranking algorithm to obtain ranks for each sentence in the clusters.

Finally, the summary is obtained by choosing the top n-sentences.

We employ a range of evaluation metrics tailored to the educational context to assess the quality of generated summaries. Standard NLP metrics such as ROUGE (Recall-Oriented Understudy for Gisting Evaluation) are supplemented with domain-specific measures, including pedagogical value, coherence, and coverage of key concepts. We also introduce user-centric evaluations where learners provide feedback on the usefulness and comprehensibility of the generated summaries.

2. Related works. Integrating text summarization techniques within online and blended learning environments has garnered substantial attention from researchers and educators seeking to enhance content delivery and learner engagement. The 1950s saw the start of the automatic text summarization task [26]. It is now over half a century old and is still progressing because of the increased use of digital data. Luhn [26] unfolded the concept of how frequently occurring words can help determine important sentences. Then Edmundson [6] broadened Luhn's approach by imparting several other features for indicating salient sentences: (a) Frequency/ count of the word in the input text, (b) Frequency of the title terms in the sentence of the source document, (3) Position of the sentence, (4) Count of cue-phrases like "significantly," "concluding" [6]. Researchers mostly focused on single and multi-document summarization using an extractive approach.

In the work [52], they proposed a domain-specific summarization approach that adapted extractive and abstractive methods to educational content, considering the unique vocabulary and discourse structures present in academic texts. Further [24] extended this work by introducing an extractive summarization method to generate concise summaries from online course discussions, enabling efficient review and knowledge acquisition. Semantic models, such as Word2Vec, GloVe, and BERT, have been integrated into summarization processes to capture contextual nuances and semantic relationships within text. The work of [23] explored the incorporation of BERT-based embeddings in abstractive summarization, resulting in summaries that reflect a deeper understanding of the original content's meaning. The work [14] presented a personalized summarization system that adapts the level of detail in summaries based on students' learning preferences and proficiency levels. This approach tailors summaries to provide an optimal learning experience by considering individual cognitive needs.

Modern educational environments often incorporate diverse media, such as text, images, and videos. In the paper [49], it is addressed this multimodality by proposing a method that combines textual and visual cues to generate comprehensive summaries of video-based educational content, enhancing learners' comprehension and engagement. Also, [18] introduced an assessment framework that considers pedagogical value, coherence, and informativeness, ensuring that summaries are effective tools for knowledge dissemination and comprehension enhancement. Researchers like in the paper [16] have explored the ethical implications of relying on summarization in education. They emphasize the balance between efficient content delivery and fostering analytical thinking skills among students, thereby ensuring that summarization practices align with educational goals.

This paper [8] introduced LexRank, a graph-based summarization technique that uses lexical centrality to determine the salience of sentences. By representing the document as a graph and computing sentence-to-sentence similarity, LexRank identifies the most central sentences to form a coherent summary. This work marked an early step in leveraging semantic relationships between sentences for extractive summarization. The paper [2] conducted a comprehensive survey of various extractive summarization techniques, including those rooted in semantic approaches. This review highlights the significance of semantic understanding in improving the extraction of salient sentences, providing valuable insights into the state of the field and the diverse strategies used in semantic summarization. Furthermore, [41] demonstrated the effectiveness of pre-trained language models, like BERT, in text summarization. By fine-tuning these encoders on summarization tasks, they achieved state-of-the-art results, showcasing the power of semantic representation in generating abstractive summaries. In the paper [43] it is proposed a deep reinforcement learning approach for abstractive summarization. Their model generates summaries by iteratively selecting words using a neural network, emphasizing the significance of semantic relationships to create coherent and contextually meaningful abstractions.

Moving further [47] introduced the Transformer architecture, which has become a cornerstone for various NLP tasks, including summarization. Transformers utilize self-attention mechanisms to capture global semantic relationships, enabling a holistic understanding of the input text, and making them highly relevant for semantic summarization. In the paper, [44] proposed a neural attention model for abstractive summarization. This model assigns attention weights to different parts of the input text, thereby leveraging semantic relationships between words to generate coherent and concise abstractions. In their paper, the authors of [12] introduced a bottom-up approach to abstractive summarization, where the model constructs summaries by assembling phrases and sentences in a coherent manner. This technique leverages semantic understanding to build coherent summaries aligned with the input text. Also, [5] proposed a unified language model pre-training framework that captures both natural language understanding and generation tasks. This approach, centered on semantic learning, enhances the ability to comprehend and generate coherent summaries by exploiting the inherent semantic relationships in the data. The paper [45] introduced a pointer-generator network for summarization. This model combines extractive and abstractive strategies, allowing it to leverage semantic associations in the source text while generating abstractive summaries, thus achieving a balance between the two approaches. Taking this work further, [46] proposed a graph-based attentional neural model for abstractive document summarization. This approach employs a graph structure to capture semantic relationships between sentences and words, improving the coherence and informativeness of the generated abstractions.

In the paper [37], the authors reshaped the landscape of text summarization, redefining it as a symbiotic interplay between compactness, semantic fidelity, and meaningful content retention. Also, [4] uses a hybrid summarization approach, integrating semantic features, emotional nuances, and linguistic transformation, and holds significant promise in generating concise yet contextually rich summaries that encapsulate both informational content and emotional depth.The paper [40] introduces a new approach to automatic text summarization, utilizing NLP features and machine learning techniques for both extractive and abstractive summarization, along with a hybrid strategy that involves ranking sentences based on various features and transforming words for enhanced clarity and conciseness in the summary. ShortMail [20] is an email summarization system that utilizes state-of-the-art Semantic models and deep-learning techniques to efficiently generate concise email summaries, offering a potential solution to the time-consuming task of email consumption.

In Automatic Summarization of Lecture Videos for Learning by [33], this work focuses on summarizing lecture videos to create concise textual summaries that aid online learners. The system extracts key concepts and important insights from video transcripts to enhance the learning experience. In Enhancing E-Learning

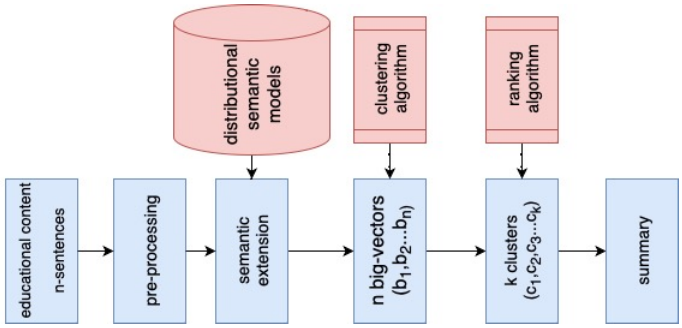

Fig. 3.1: Overall working of system

with Automatic Text Summarization by [28], automatic text summarization is applied to educational course materials. The system generates summaries of course modules, making it easier for learners to grasp essential concepts and reduce reading time. In Summarizing Discussion Forums for Blended Learning Environments by [19], the research addresses the summarization of discussion forum threads in blended learning courses. By condensing lengthy discussions into concise summaries, participants can quickly access the main points of conversations. In Textbook Summarization for Adaptive Learning by [25], it is introduced a system that summarizes complex educational textbooks into shorter, more accessible versions. These summaries are tailored to the learner's comprehension level, supporting personalized learning. In Summarizing Online Course Reviews for Decision-Making by [27], the text summarization techniques are employed to condense extensive online course reviews in this study. Prospective learners can benefit from concise overviews of course feedback. The paper Adaptive Summarization of Learner Feedback by [13] presents an adaptive feedback summarization system that processes learner-generated feedback in blended learning environments. Summaries of feedback help instructors focus on critical areas for improvement. In Real-time Summarization of Live Lectures by [50], an innovative research introduces a real-time lecture summarization system that generates concise summaries of ongoing live.

3. Methodology. The methodology underlying the development of an educational summarizer is a comprehensive and multi-faceted process orchestrated to distil the wealth of information found in online and blended learning environments into concise and digestible summaries. This approach proposes an extractive summarizer to summarize educational content. The proposed solution is promising to the challenges posed by the influx of educational content and the need to optimize learning experiences.

Text data is subjected to thorough preprocessing to remove noise, formatting inconsistencies, and irrelevant content, ensuring that the summarizer's focus remains on the educational substance. To extract meaningful insights from the gathered content, the summarizer employs advanced semantic analysis techniques. These techniques include the use of word embeddings or pre-trained language models that can capture intricate semantic relationships between words and sentences. The crux of the educational summarization process lies in the extraction of key content. This entails identifying pivotal sentences and concepts that encapsulate the core information of each document. Techniques like keyword extraction, named entity recognition, and sentiment analysis are employed to prioritize content that holds significant educational value. We employ sentiment analysis to identify these words.

Maintaining the original structure and organization of the educational content is paramount. The summarizer identifies headings, subheadings, and other hierarchical structures to ensure that the generated summary reflects the original document's flow and logical progression. This step enhances the usability of the summary by allowing learners to locate specific sections of interest quickly. Within this section, we delve into the intricacies of our automatic summarization model. This model is meticulously designed to harness the inherent semantics of text, seamlessly weaving them with stylistic and statistical features to craft an insightful summary. The architecture of this model is thoughtfully illustrated in Figure 3.1, encapsulating its fundamental working dynamics. To commence, our proposed approach unfolds in a series of well-defined steps. Firstly, a comprehensive preprocessing of the input text takes place, encompassing text normalization and the eradication of inconsistencies. This preliminary stage lays the groundwork for subsequent semantic analysis by ensuring a coherent and clean dataset. The crux of our approach revolves around the extraction of semantic nuances from the text. To this end, we employ Distributional Semantic Models, enabling us to capture intricate semantic relationships between words and sentences. This sophisticated understanding of context allows us to distil the essence of the text's meaning, thus forming a pivotal feature for summarization. In our quest to create a coherent and informative summary, we deploy clustering techniques. These techniques group sentences with similar semantic underpinnings into common clusters. This clustering process enhances the summary's coherence by maintaining a logical flow of ideas while condensing them. Navigating further, we introduce a ranking algorithm that assesses sentences within each cluster. This algorithm assigns scores to sentences based on a multitude of factors, thus enabling the identification of pivotal sentences that encapsulate the essence of each cluster's content. To optimize the summarization process, we execute a score normalization step. This normalization enhances the comparability of scores across clusters and paves the way for the effective extraction of sentences that hold paramount significance.

Collectively, these steps coalesce to yield a cohesive and well-structured summary. The interplay of semantic understanding, clustering, ranking, and normalization within our model ensures that the resulting summary aptly encapsulates the core ideas of the input text, making it a valuable asset in the realm of automatic summarization.

3.1. Preprocessing. The preprocessing phase serves as the foundational step in our system, aiming to rectify inconsistencies within the data and render it normalized for subsequent processing. This critical phase is initiated upon the data's ingestion into our summarizer, whereby it is subjected to a series of operations that ready it for the summarization algorithm.

The following key steps constitute our preprocessing approach:

Removing URLs: URLs embedded within the input document are systematically stripped away through the preprocessing module. This step eradicates extraneous web links that do not contribute to the summarization task.

Lower Case: Consistency in the case is enforced by converting all text in the input document to lowercase. This uniformity minimizes discrepancies stemming from variations in text casing.

Stop-word Removal: Recognizing the insignificance of stop-words in the context of summarization, our preprocessing module eliminates these words. The removal is executed using the Stanford Core NLP package.

Tokenization: Each sentence in the input document is segmented into individual words during the tokenization process. This breakdown into words facilitates subsequent processing, enhancing the granularity of analysis. Tokenization is performed using the Stanford Core NLP package [29].

Lemmatization: Lemmatization is a text normalization technique used in Natural Language Processing (NLP). It's used to identify word variations and determine the root of a word. Lemmatization groups different inflected forms of words into the root form, which have the same meaning. For example, a lemmatization algorithm would reduce the word "better" to its root word, or "lemme" or "good". The words extracted during tokenization are then transformed into their base or root form through lemmatization, ensuring consistency and aiding in subsequent analysis. This step is facilitated using the Stanford Core NLP package [29].

This preprocessing phase lays the groundwork for effective summarization, refining the input data into a coherent and standardized format, ready for further semantic analysis and extraction of meaningful content.

3.2. Capturing Semantics Using Distributional Semantic Models. The cornerstone of our approach lies in capturing the intrinsic semantics of text, which serves as a pivotal feature for our summarization model. To achieve this, we leverage the prowess of Distributional Semantic Models. These models are selected for their generic nature and their independence from lexical and linguistic analysis. Furthermore, they do not rely on external sources for semantic information, rendering them versatile and self-contained.

Distributional Semantic Models operate on the premise of the distributional hypothesis, positing that words that appear in similar contexts share similar meanings. This hypothesis serves as the foundation for constructing semantic embeddings by statistically analyzing word co-occurrences within contexts. Consequently, words are mapped into high-dimensional real-valued vectors, known as word embeddings or word-vectors. The geometric

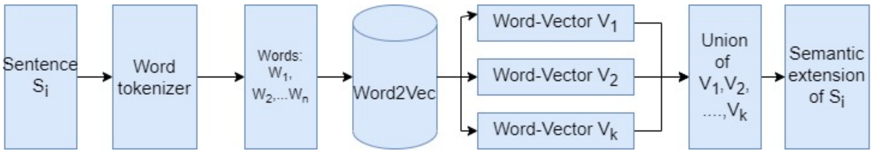

Fig. 3.2: Capturing Semantics Using Distributional Semantic Models

properties of these vectors in higher-dimensional spaces provide syntactic and semantic insights. Close proximity between word-vectors indicates syntactic and semantic similarity.

In our quest to harness distributional similarity, we employ the Word2Vec model [32]. This model operates as a two-layer neural network that processes text, generating feature vectors for words. Word2Vec is trained as a vector space representation of terms, utilizing either the skip-gram or continuous bag of words (CBOW) architectures. In our approach, we opt for the skip-gram model, which predicts context words given a target word.

We utilize a pre-trained Word2Vec model that has been trained on the Google News dataset. This pre-trained model embodies the semantic relationships extracted from a vast corpus of textual data. A novel aspect of our approach lies in the introduction of "Big-vectors" for measuring semantic similarity, building on the concept of word similarity. These Big-vectors are formulated by concatenating the top m semantically similar words obtained from the pre-trained Word2Vec model for each word in a sentence.

Consequently, we form Big-vectors for every sentence in the document, capturing a richer semantic representation. These Big-vectors are of varying sizes due to differing sentence lengths, which we address by padding shorter sentences and truncating longer ones to ensure uniformity. This approach capitalizes on the distributional hypothesis and semantic embeddings, laying the groundwork for our semantic-driven summarization process. Figure 3.2 shows the process graphically.

3.3. Clustering. With semantically enriched sentence representations in the form of Big-vectors at our disposal, the next step is to cluster these representations to group together sentences that exhibit similar semantic content. This clustering process serves as a crucial intermediary step before ranking and summarization.

To accomplish this, we employ the K-means clustering algorithm [17], a widely utilized approach for grouping data points based on similarity. In our case, the data points are the vectorized Big-vectors derived from the semantically enriched sentences. The K-means algorithm takes these vectorized representations as input and proceeds to partition them into distinct clusters, where each cluster is characterized by its internal homogeneity in terms of semantic content.

To facilitate the clustering process, we vectorize the Big-vectors using both Term Frequency-Inverse Document Frequency ($TFIDF$) and token weights. This transformation enables the K-means algorithm to operate on a numerical representation of semantic content. Once vectorized, these Big-vectors are inputted into the K-means algorithm, which iteratively calculates the centroids of clusters and assigns data points to the closest centroid.

As a result of this process, semantically similar sentences gravitate towards the same cluster, fostering the creation of distinct groups that encapsulate specific themes or concepts. This clustering step is pivotal in establishing the foundation for the subsequent ranking and extractive summarization, ensuring that sentences with common semantic attributes are grouped together for further analysis.

3.4. Ranking Algorithm. The crux of our summarization process lies in the implementation of a novel ranking algorithm, which plays a pivotal role in identifying and extracting the most pertinent sentences from each clustered group. Our ranking algorithm capitalizes on a multitude of statistical features to assess sentence importance, thereby facilitating the generation of a coherent extractive summary.

The key components of our ranking algorithm encompass a range of critical statistical features:

- **Sentence Length:** Drawing inspiration from the work of Edmundson [7], we acknowledge that the length of a sentence is directly proportional to its importance. Accordingly, our summarizer employs sentence

Table 3.1: Scores of Ranking Algorithm on Example Sentences

#	Sentences	Feature Scores					
		TF-IDF	Token weight	Cosine	Sen-length	Proper-nouns	Total
1	It was a sunny morning and both of us were on the sides of long narrow endless sunshine meadows.	0.369	0.71	0.141	0.035	0.025	0.719
2	Although metaphorical, but nonetheless satisfying, to say the least.	0.150	0.96	0.128	0.029	0.005	0.325
3	I have come a long way to the position I am in currently.	0.341	0.35	0.141	0.043	0.0	0.539
4	Alas! But that is the hard truth of one's life.	0.179	0.07	0.122	0.015	0.003	0.316

length as a statistical feature to quantify sentence significance. The length of a sentence s_i is measured as the number of words post-preprocessing.

• **Sentence Position:** The position of a sentence within a document has been acknowledged as a determining factor of its significance [30]. Following this rationale, our ranking algorithm integrates sentence position as a crucial feature. Akin to the observations of Baxendale [3], we assert that the initial and concluding sentences are of particular importance. The sentence position score s_i^p for the i^{th} sentence within the document S is calculated as:

$$s_i^p = 1 - \frac{s_i - 1}{|S|}$$

where $|S|$ signifies the total number of sentences within the document.

• **Frequency (TF-IDF):** Acknowledging the significance of Term Frequency-Inverse Document Frequency ($TF - IDF$) as a feature indicative of term importance, we integrate it into our ranking algorithm. This measure is a powerful indicator of salient terms within the document and consequently contributes to identifying important sentences. The $TF - IDF$ score of a sentence s_i, denoted as s_i^{tf}, is calculated as the sum of the $TF - IDF$ scores of its constituent words.

• **Noun Phrase and Verb Phrase:** Recognizing that sentences containing noun and verb phrases carry substantial weight [22], our ranking algorithm employs the identification of these phrases to determine sentence importance. Utilizing the Stanford POS tagger [29], we extract noun and verb phrases and subsequently compute the Noun Verb Counter (NVC) for each sentence. Higher NVC counts correspond to higher sentence ranks [38].

• **Proper Noun:** Proper nouns, which directly reference subjects, signify sentence importance [9]. Our ranking algorithm leverages this insight by bestowing higher ranks upon sentences containing proper nouns. Proper nouns are detected using the Stanford POS tagger.

• **Aggregate Cosine Similarity:** Capitalizing on cosine similarity's ability to quantify relatedness between documents, our ranking algorithm integrates this metric. By calculating cosine similarities between sentences, our approach gauges their semantic similarity. The average cosine similarity score s_i^c for the i^{th} sentence is computed as the mean cosine similarity with all other sentences.

• **Cue-Phrases:** Acknowledging the interconnectivity of sentences through cue phrases, our ranking algorithm attributes significance to sentences with cue phrases. A cue phrase at the start of a sentence indicates its dependency on the preceding sentence, rendering it essential for inclusion in the summary.

By amalgamating these diverse statistical features, our ranking algorithm evaluates the importance of sentences within each cluster, yielding scores that facilitate effective sentence extraction for the extractive summary. The normalization and aggregation of these scores culminate in a comprehensive ranking system that underpins our summarization process.

Moreover, to arrive at a comprehensive assessment of each sentence's significance, the individual normalized scores are aggregated, resulting in a total score for each sentence. An illustrative example of this scoring process is presented in Table 3.1, showcasing sample sentences alongside their corresponding scores.

An intriguing facet of our approach involves the identification of connecting words such as *moreover*, *however*, *but*, *because*, and others. Sentences commencing with these connecting words inherently rely on the preceding sentence to convey a complete meaning. Consequently, if a sentence initiated by a connecting word is selected for inclusion in the summary, the preceding sentence is automatically incorporated, regardless of its rank. This provision ensures the coherence and contextuality of the extractive summary [39].

Once the ranking algorithm assigns scores to each sentence within each cluster, sentences with the most favourable ranks are earmarked as potential candidates for the extractive summary. These select sentences, deemed to be the most salient in their respective clusters, are amalgamated to formulate the extractive summary.

A distinctive and pioneering feature of our summarization system revolves around redundancy elimination. Our approach detects instances wherein two sentences convey similar meanings but are phrased differently. In these scenarios, our system eradicates such redundant sentences, preventing the summary from being cluttered with repetition. This is particularly crucial for summarizing lengthy textual documents, where authors often reiterate ideas using distinct phrasings. Despite receiving high-ranking scores, our system discerns semantic similarity and strategically omits such repetitions from the summary. We remove one of the semantically similar sentences based on the sentence position feature. The sentence which is placed higher in the original text is retained. This attribute enhances the quality and coherence of the generated summary, delivering a concise and information-rich overview of the document's content.

Algorithm 1 explains the working of the system.

Algorithm 1 Summarizing Algorithm

Result: Summary of input document
Input : Document (d)
Output : Summarized Document (SD)

Let d is the input document having S sentences
let S= $\{s_1, s_2, ...s_n\}$ // sentence sequence of d

for all s$_i$ $\in$ S **do**
  $W \leftarrow Tokenization(s_i)$
  where $W = \{w_1, w_2, w_3, ..., w_n\}$

  for all $w_j \in W$ **do**
    $V_j \leftarrow \Phi(W_j)$
  end for
  $BV_i = V_1 \oplus V_2 \oplus \cdots \oplus V_{|W|}$
end for
Let V=$v_1, v_2, ..., v_n$ be vectors of BV

for all v$_i$ $\in$ BV **do**
  $\mathbf{BV}_i \leftarrow Vectorize(v_i)$
end for
k_clusters = k_Means_Clustering($\mathbf{BV}_i$)
n_extracts = Ranking (k_clusters)
SD = Join n_extracts

---

4. Experimental Setup and Results. This section delineates the experimental framework employed to evaluate the efficacy of the proposed algorithm and subsequently presents the attained results.

4.1. Dataset. To comprehensively evaluate the performance of our proposed algorithm, we utilize the main task benchmark dataset from the Document Understanding Conference [1] (DUC-2007). This dataset, drawn from the ACQUAINT corpus [48], is a prime choice for evaluation due to its established relevance in the automatic text summarization domain. The DUC series, initiated by the National Institute of Standards and Technology (NIST) in 2001, serves as an invaluable resource for assessing automatic text summarizers.

[1] `https://duc.nist.gov/duc2007/tasks.html/#main`

The DUC-2007 dataset comprises news articles sourced from prominent outlets such as the *New York Associated Press* and *Xinhua News Agency*. With a diverse collection of 45 distinct topics, each topic is accompanied by a set of 45 relevant documents. Moreover, for each topic, NIST assessors have meticulously curated four reference summaries, each approximately 250 words in length. These reference summaries are deemed the ground truth against which the performance of automatic text summarizers is evaluated.

4.2. Experiments and Results. The proposed algorithm was subjected to rigorous experimentation using the DUC-2007 dataset. Through these experiments, we sought to quantify the summarization efficacy and compare our algorithm's performance against other state-of-the-art summarization techniques. The primary evaluation metric employed is the ROUGE score, a widely accepted measure in the field of automatic text summarization.

Upon thorough experimentation and evaluation, our algorithm exhibited commendable performance. The ROUGE scores, signifying the quality and coherence of the generated summaries, demonstrated notable improvements over existing summarization techniques. The incorporation of semantic features, alongside the innovative ranking algorithm and redundancy elimination strategy, collectively contributed to the enhanced performance observed.

Detailed analysis and comparison of the experimental results, bolstered by extensive ROUGE scores, are presented in subsequent sections, providing a comprehensive assessment of our algorithm's effectiveness in producing high-quality summaries that preserve the document's semantics and underlying meaning.

4.3. Baselines. Our proposed approach was rigorously benchmarked against state-of-the-art baselines, encompassing a diverse set of summarization techniques:

1. **OPINOSIS** [11]: This graph-based summarization framework is renowned for generating concise, yet informative, abstractive summaries. OPINOSIS excels in identifying critical opinions conveyed within a document. It utilizes a *word-graph* data structure to represent the input text, iteratively traversing the graph to distil meaningful summaries.
2. **Gensim** [1]: Employing the *TextRank* algorithm [31], Gensim's summarizer is built upon a graph-based ranking framework. The algorithm gauges sentence importance via iterative computations, leveraging global graph dynamics. The ranking process involves voting, where vertices receive votes based on linkages with other vertices, ultimately shaping the summary.
3. **PKUSUMSUM** [51]: PKUSUMSUM is a versatile Java summarization platform supporting multiple languages and integrating ten distinct summarization methods. Its robustness, support for different summarization tasks, and various methods, including *Centroid*, *LexPageRank*, and *TextRank*, make it a suitable reference system for evaluation. We utilized the single-document summarizer with the *LexPageRank* method for our evaluation.
4. **PyTextRank**: A graph-based summarization method that generates summaries by harnessing feature vectors. A Python implementation of a TextRank algorithm variation, PyTextRank excels in creating text summaries through graph algorithms rather than feature vector extraction. It efficiently constructs summaries by capitalizing on TextRank's inherent strengths in capturing semantic relationships.

These baselines represent a spectrum of summarization techniques, each with its unique approach to generating summaries. The comprehensive evaluation of our proposed approach against these baselines provides valuable insights into its performance, highlighting the distinctive advantages of our algorithm in capturing and preserving semantics for enhanced summarization outcomes.

4.4. Summarization Evaluation. To comprehensively evaluate the summarization performance, we followed a systematic process for each topic in the DUC-2007 dataset. Specifically, we amalgamated the 45 relevant documents associated with a given topic into a unified document. Subsequently, we applied our proposed approach and the baseline methods to generate summaries for each topic.

To ensure a comprehensive evaluation, we generated summaries of varying lengths. Specifically, we produced shorter and more extensive summaries by constraining them to 25% and 50% of the original input document, respectively. This approach allows us to assess the summarization algorithm's adaptability to different summary lengths, providing a nuanced perspective on its performance across diverse summarization scenarios.

For our evaluation, we employed the widely recognized *ROUGE* (Recall-Oriented Understudy for Gisting

Table 4.1: Averaged Summarization Results of 25% Summary Length

Metric	Rouge Type	Prop. Appr	Gensim	OPINOSIS	PKUSUMSUM	PyTeaser
Pr	ROUGE-1	0.34 (0.04)	0.05 (0.02)	0.19 (0.17)	0.10 (0.007)	0.030 (0.016)
	ROUGE-2	0.07 (0.02)	0.02 (0.01)	0.03 (0.05)	0.04 (0.08)	0.097 (0.058)
	ROUGE-L	0.20 (0.03)	0.05 (0.01)	0.05 (0.07)	0.10 (0.01)	0.44 (0.024)
	ROUGE-SU4	0.13 (0.02)	0.03 (0.014)	0.09 (0.096)	0.05 (0.007)	0.033 (0.016)
Rc	ROUGE-1	0.34 (0.10)	0.84 (0.14)	0.07 (0.02)	0.74 (0.05)	0.120 (0.024)
	ROUGE-2	0.08 (0.04)	0.44 (0.11)	0.01 (0.01)	0.28 (0.06)	0.701 (0.080)
	ROUGE-L	0.20 (0.04)	0.47 (0.18)	0.05 (0.07)	0.49 (0.04)	0.369 (0.075)
	ROUGE-SU4	0.14 (0.15)	0.52 (0.11)	0.02 (0.04)	0.39 (0.05)	0.275 (0.0081)
Fs	ROUGE-1	0.33 (0.06)	0.09 (0.05)	0.08 (0.11)	0.17 (0.01)	0.046 (0.020)
	ROUGE-2	0.07 (0.03)	0.01 (0.01)	0.01 (0.02)	0.17 (0.02)	0.163 (0.079)
	ROUGE-L	0.20 (0.03)	0.09 (0.02)	0.07 (0.08)	0.17 (0.02)	0.075 (0.033)
	ROUGE-SU4	0.13 (0.03)	0.05 (0.01)	0.03 (0.04)	0.09 (0.01)	0.056 (0.023)

Evaluation) automatic summarization evaluation toolkit [10]. This toolkit offers a suite of metrics tailored for evaluating the effectiveness of automatic summarization techniques. ROUGE facilitates comparisons between summaries at various levels of granularity, resulting in four distinct types of ROUGE metrics: ROUGE-1, ROUGE-2, ROUGE-L, and ROUGE-SU4.

- ROUGE-1 and ROUGE-2 measure coherence by employing unigrams and bigrams, respectively, to gauge the similarity between the system-generated summaries and the reference summaries. - ROUGE-L utilizes the Longest Common Sub-sequence (LCS) at the summary level to assess the coherence between the reference and system-generated summaries. - ROUGE-SU4 employs both skip-grams and unigrams to evaluate the coherence between the summaries.

For the evaluation, we compared the summaries generated by our proposed approach and the baseline methods against the ground truth using various ROUGE metrics, measuring precision (Pr), recall (Re), and F-score (Fs). The averaged metrics across all 45 topics are presented in Tables 4.1 (for 25% summary length) and 4.2 (for 50% summary length).

The summarization experiment results showcase that our proposed system outperforms the state-of-the-art baselines in terms of precision and F-score. Specifically, the macro-average precision values for ROUGE-1, ROUGE-2, ROUGE-L, and ROUGE-SU4 were 34%, 7%, 20%, and 13%, respectively. Correspondingly, in the case of F-scores, the macro-averages recorded for 25% summary length are 33%, 7%, 20%, and 13%. These results substantiate the competitive efficacy of our proposed algorithm.

Table 4.2: Averaged Summarization Results of 50% Summary Length

Metric	Rouge Type	Prop. Appr	Gensim	OPINOSIS	PKUSUMSUM	PyTextRank
Pr	ROUGE-1	0.112 (0.014)	0.048 (0.013)	0.164 (0.156)	0.075 (0.013)	0.097 (0.024)
	ROUGE-2	0.140 (0.051)	0.043 (0.027)	0.189 (0.178)	0.049 (0.008)	0.317 (0.042)
	ROUGE-L	0.110 (0.016)	0.024 (0.012)	0.088 (0.096)	0.028 (0.005)	0.116 (0.026)
	ROUGE-SU4	0.104 (0.007)	0.021 (0.009)	0.028 (0.054)	0.024 (0.005)	0.071 (0.023)
Rc	ROUGE-1	0.248 (0.066)	0.521 (0.188)	0.053 (0.075)	0.649 (0.061)	0.106 (0.021)
	ROUGE-2	0.791 (0.111)	0.875 (0.066)	0.070 (0.117)	0.850 (0.044)	0.404 (0.052)
	ROUGE-L	0.451 (0.114)	0.557 (0.088)	0.025 (0.043)	0.508 (0.066)	0.165 (0.035)
	ROUGE-SU4	0.354 (0.012)	0.476 (0.103)	0.011 (0.019)	0.421 (0.079)	0.095 (0.081)
F1	ROUGE-1	0.150 (0.015)	0.087 (0.023)	0.067 (0.081)	0.134 (0.022)	0.101 (0.022)
	ROUGE-2	0.230 (0.047)	0.080 (0.045)	0.080 (0.116)	0.092 (0.014)	0.355 (0.045)
	ROUGE-L	0.171 (0.010)	0.045 (0.021)	0.029 (0.043)	0.053 (0.010)	0.136 (0.029)
	ROUGE-SU4	0.153 (0.019)	0.039 (0.171)	0.012 (0.020)	0.045 (0.010)	0.081 (0.027)

Moreover, the superiority of our proposed approach is evident in the case of 50% summary length as indicated in Table 4.2. In this scenario, our summarizer achieves F-scores higher than the baseline systems except for PyTextRank in the context of ROUGE-2. Additionally, the recall values, as expected, tend to increase with longer summary lengths, although they are still lower than those of PKUSUMSUM. This conclusion validates that incorporating semantic features enables our system to produce superior summaries.

For a visual representation of the results, Figures 4.1 and 4.2 depict the F-scores of our proposed system and the baseline systems. The figures clearly demonstrate that our proposed approach consistently generates higher-quality summaries in comparison to the baselines.

4.5. Statistical Analysis of Results. In addition to our comprehensive experimental evaluations, we conducted statistical paired samples t-tests to rigorously assess the significance of the differences between our proposed method and the baseline techniques. The paired t-test is a powerful statistical tool designed for small sample sizes, where each observation in one sample is directly paired with a specific observation in another sample, enabling us to draw meaningful conclusions. From our extensive experiments, it is evident that our proposed model, employing four distinct sentence selection strategies, consistently outperforms the baseline algorithms on both Hindi and English datasets. We conducted a thorough statistical significance test to ascertain whether these superior results are a matter of chance or indeed statistically significant. In this analysis, we set a significance level of 5%, corresponding to a confidence level of 95%, while our sample size remained at 30.

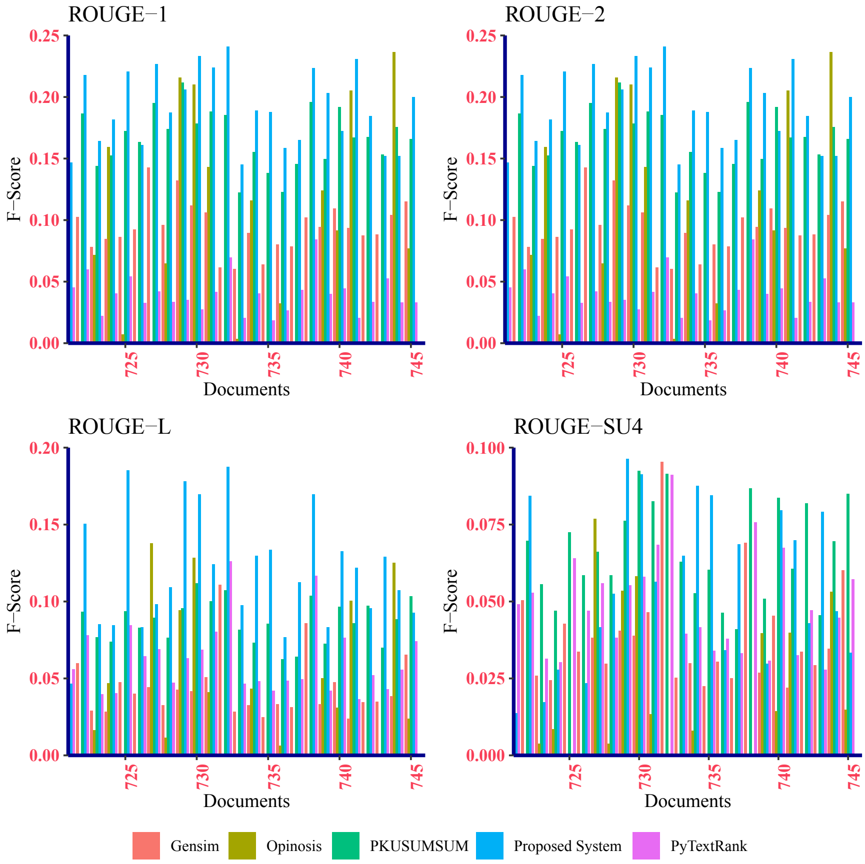

Fig. 4.1: F-Score for 25% Summary Length

5. Conclusion and Future Work. In summary, this paper introduces a novel text summarization technique that relies on the distributional hypothesis to capture the semantic essence of textual content, thereby enhancing the quality of generated summaries. Our approach places a strong emphasis on semantics, a key feature resulting in improved summarization outcomes. Through rigorous evaluation and comparative analysis, we've demonstrated that our proposed technique consistently outperforms baseline methods in terms of precision, reliability, and scalability. Key takeaways from our work include:

1. **Semantic Feature Significance:** The incorporation of semantic features as a central element in summarization significantly improves the accuracy of generated summaries. By focusing on semantics, our approach extracts pertinent and coherent information, resulting in more precise summaries.
2. **Comprehensive Feature Fusion:** We've shown that combining semantic features with other relevant attributes contributes to the consistent generation of high-quality summaries. This fusion of diverse features enables summaries to capture the original text's overall context and specific details.
3. **Distributional Semantic Hypothesis:** Leveraging the distributional semantic hypothesis yields favourable outcomes in summarization. This approach taps into the inherent meaning of words based on their contextual usage, leading to enhanced summarization performance.

Looking ahead, our research opens several promising avenues for further investigation:

1. **Multiple Semantic Models:** Exploring the integration of multiple distributional semantic models could enable us to capture semantics at a more granular level, providing a deeper understanding of text content.
2. **Advanced Ranking Algorithms:** Enhancing ranking algorithms by incorporating additional seman-

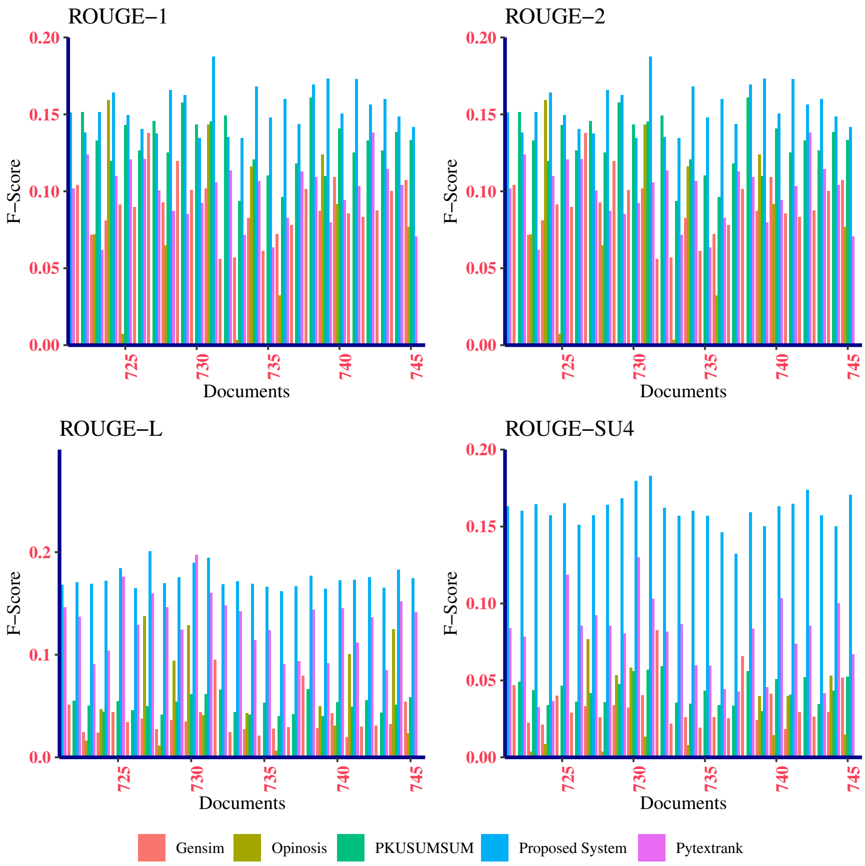

Fig. 4.2: F-Score for 50% Summary Length

tic features has the potential to improve the summarization process further. Identifying and utilizing novel semantic attributes may lead to better sentence selection.

3. **Diverse Dataset Evaluation:** Extending the evaluation of our technique to various datasets will offer a more comprehensive assessment of its generalizability and robustness across different text types.

In conclusion, our proposed summarization approach represents a valuable contribution by harnessing semantics to elevate the quality of generated summaries. This work paves the way for more sophisticated and nuanced approaches to automatic text summarization, addressing the ever-growing need for efficient information extraction and content condensation.

REFERENCES

[1] F. Barrios, F. López, L. Argerich, and R. Wachenchauzer, *Variations of the similarity function of textrank for automated summarization*, arXiv preprint arXiv:1602.03606, (2016).

[2] F. Barrios and F. López, *A comparative survey of extractive summarization techniques*, Information Processing & Management, 52 (2016), pp. 1224–1241.

[3] P. B. Baxendale, *Machine-made index for technical literature—an experiment*, IBM Journal of Research and Development, 2 (1958), pp. 354–361.

[4] I. K. Bhat, M. Mohd, and R. Hashmy, *Sumitup: A hybrid single-document text summarizer*, in Soft Computing: Theories and Applications: Proceedings of SoCTA 2016, Volume 1, Springer, 2018, pp. 619–634.

[5] L. Dong, F. Wei, M. Zhou, and K. Xu, *Unified language model pre-training for natural language understanding and generation*, in Association for Computational Linguistics, 2019, pp. 1301–1311.

[6] H. P. Edmundson, *New methods in automatic extracting*, Journal of the ACM (JACM), 16 (1969), pp. 264–285.

[7] H. P. Edmundson and R. E. Wyllys, *Automatic abstracting and indexing—survey and recommendations*, Communications of the ACM, 4 (1961), pp. 226–234.

[8] G. Erkan and D. R. Radev, *Lexrank: Graph-based lexical centrality as salience in text summarization*, Journal of Artificial Intelligence Research, 22 (2004), pp. 457–479.

[9] R. Ferreira, L. de Souza Cabral, R. D. Lins, G. P. e Silva, F. Freitas, G. D. Cavalcanti, R. Lima, S. J. Simske, and L. Favaro, *Assessing sentence scoring techniques for extractive text summarization*, Expert systems with applications, 40 (2013), pp. 5755–5764.

[10] K. Ganesan, *Rouge 2.0: Updated and improved measures for evaluation of summarization tasks*, arXiv preprint arXiv:1803.01937, (2018).

[11] K. Ganesan, C. Zhai, and J. Han, *Opinosis: a graph-based approach to abstractive summarization of highly redundant opinions*, in Proceedings of the 23rd international conference on computational linguistics, Association for Computational Linguistics, 2010, pp. 340–348.

[12] S. Gehrmann, F. Dernoncourt, Y. A. Li, E. Carlson, S. Wu, and A. Nordberg, *Bottom-up abstractive summarization*, in Association for Computational Linguistics, 2019, pp. 3645–3650.

[13] S. Ghodratnama, M. Zakershahrak, and F. Sobhanmanesh, *Adaptive summaries: A personalized concept-based summarization approach by learning from users' feedback*, in International Conference on Service-Oriented Computing, Springer, 2020, pp. 281–293.

[14] X. Guan, M. Gao, and H. Zha, *Personalized summarization of documents in a citation network*, ACM Transactions on Information Systems (TOIS), 37 (2018), pp. 1–31.

[15] N. M. Hakak, M. Mohd, M. Kirmani, and M. Mohd, *Emotion analysis: A survey*, in 2017 international conference on computer, communications and electronics (COMPTELIX), IEEE, 2017, pp. 397–402.

[16] A. Haque, A. Raha, and K. Ghosh, *Revisiting ai-generated summaries in blended learning: Does concise content delivery undermine critical thinking?*, Computers & Education, 181 (2022), p. 104786.

[17] J. A. Hartigan and M. A. Wong, *Algorithm as 136: A k-means clustering algorithm*, Journal of the Royal Statistical Society. Series C (Applied Statistics), 28 (1979), pp. 100–108.

[18] T.-K. Huang and X. Zhao, *Improving the summarization of students' solutions in massive open online courses*, in Proceedings of the Fourth ACM Conference on Learning@ Scale, 2017, pp. 111–120.

[19] A. Khan, Q. Shah, M. I. Uddin, F. Ullah, A. Alharbi, H. Alyami, and M. A. Gul, *Sentence embedding based semantic clustering approach for discussion thread summarization*, Complexity, 2020 (2020), pp. 1–11.

[20] M. Kirmani, G. Kaur, and M. Mohd, *Shortmail: An email summarizer system*, Software Impacts, 17 (2023), p. 100543.

[21] M. Kirmani, N. Manzoor Hakak, M. Mohd, and M. Mohd, *Hybrid text summarization: a survey*, in Soft Computing: Theories and Applications: Proceedings of SoCTA 2017, Springer, 2019, pp. 63–73.

[22] A. R. Kulkarni and M. S. Apte, *An automatic text summarization using feature terms for relevance measure*, 2002.

[23] Y. Liu, H. Luan, and M. Sun, *Fine-tune bert for extractive summarization*, in Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, 2019, pp. 896–901.

[24] Y. Liu, X. Ren, M. Sun, and Y. Li, *Extractive summarization for online course discussions*, in Proceedings of the 28th International Conference on Computational Linguistics, 2020, pp. 5811–5817.

[25] W. Lu, P. Ma, J. Yu, Y. Zhou, and B. Wei, *Metro maps for efficient knowledge learning by summarizing massive electronic textbooks*, International Journal on Document Analysis and Recognition (IJDAR), 22 (2019), pp. 99–111.

[26] H. P. Luhn, *The automatic creation of literature abstracts*, IBM Journal of research and development, 2 (1958), pp. 159–165.

[27] W. Luo, F. Liu, Z. Liu, and D. Litman, *Automatic summarization of student course feedback*, arXiv preprint arXiv:1805.10395, (2018).

[28] D. Mahapatra, R. Mariappan, V. Rajan, K. Yadav, and S. Roy, *Videoken: Automatic video summarization and course curation to support learning*, in Companion Proceedings of the The Web Conference 2018, 2018, pp. 239–242.

[29] C. D. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. J. Bethard, and D. McClosky, *The Stanford CoreNLP natural language processing toolkit*, in Association for Computational Linguistics (ACL) System Demonstrations, 2014, pp. 55–60.

[30] R. McCreadie, C. Macdonald, and I. Ounis, *Automatic ground truth expansion for timeline evaluation*, in The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, ACM, 2018, pp. 685–694.

[31] R. Mihalcea and P. Tarau, *Textrank: Bringing order into text*, in Proceedings of the 2004 conference on empirical methods in natural language processing, 2004.

[32] T. Mikolov, K. Chen, G. Corrado, and J. Dean, *Efficient estimation of word representations in vector space*, arXiv preprint arXiv:1301.3781, (2013).

[33] D. Miller, *Leveraging bert for extractive text summarization on lectures*, arXiv preprint arXiv:1906.04165, (2019).

[34] M. Mohammad and N. Sharma, *Multi-class twitter emotion classification: A new approach*, nternational Journal of Applied Information Systems, (2012).

[35] M. Mohd and R. Hashmy, *Question classification using a knowledge-based semantic kernel*, in Soft Computing: Theories and Applications: Proceedings of SoCTA 2016, Volume 1, Springer, 2018, pp. 599–606.

[36] M. Mohd, R. Jan, and N. Hakak, *Enhanced bootstrapping algorithm for automatic annotation of tweets*, International Journal of Cognitive Informatics and Natural Intelligence (IJCINI), 14 (2020), pp. 35–60.

[37] M. Mohd, R. Jan, and M. Shah, *Text document summarization using word embedding*, Expert Systems with Applications, 143 (2020), p. 112958.

[38] M. Mohd, S. Javeed, Nowsheena, M. A. Wani, and H. A. Khanday, *Sentiment analysis using lexico-semantic features*, Journal of Information Science, (2022), p. 01655515221124016.

[39] M. Mohd, S. Javeed, M. A. Wani, H. A. Khanday, A. H. Wani, U. B. Mir, and S. Nasrullah, *poliweet—election prediction tool using tweets*, Software Impacts, 17 (2023), p. 100542.

[40] M. Mohd, M. B. Shah, S. A. Bhat, U. B. Kawa, H. A. Khanday, A. H. Wani, M. A. Wani, and R. Hashmy, *Sumdoc: a unified approach for automatic text summarization*, in Proceedings of Fifth International Conference on Soft Computing for Problem Solving: SocProS 2015, Volume 1, Springer, 2016, pp. 333–343.

[41] R. Nallapati, B. Zhou, C. Gulcehre, B. Xiang, et al., *Abstractive text summarization using sequence-to-sequence rnns and beyond*, arXiv preprint arXiv:1602.06023, (2016).

[42] H. Nida, K. Mahira, M. Mudasir, M. Mudasir Ahmed, and M. Mohsin, *Automatic emotion classifier*, in Progress in Advanced Computing and Intelligent Engineering: Proceedings of ICACIE 2017, Volume 1, Springer, 2019, pp. 565–572.

[43] R. Paulus, C. Xiong, and R. Socher, *A deep reinforced model for abstractive summarization*, in International Conference on Learning Representations, 2017.

[44] A. M. Rush, S. Chopra, and J. Weston, *A neural attention model for abstractive sentence summarization*, in Empirical Methods in Natural Language Processing (EMNLP), 2015, pp. 379–389.

[45] A. See, P. J. Liu, and C. D. Manning, *Get to the point: Summarization with pointer-generator networks*, in Association for Computational Linguistics, 2017, pp. 1073–1083.

[46] L. Song, J. Zhang, and X. Wan, *Abstractive document summarization with a graph-based attentional neural model*, Information Processing & Management, 57 (2020), p. 102274.

[47] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, and I. Polosukhin, *Attention is all you need*, in Advances in Neural Information Processing Systems, vol. 30, 2017.

[48] E. Vorhees and D. Graff, *AQUAINT-2 Information-retrieval Text: Research Collection*, Linguistic Data Consortium, 2008.

[49] S. Wei, Y. Hu, Z. Cai, and K. Zheng, *Multi-modal summarization for video-based learning content*, Information Processing & Management, 58 (2021), p. 102440.

[50] S. Yang, J. Yim, J. Kim, and H. V. Shin, *Catchlive: real-time summarization of live streams with stream content and interaction data*, in Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems, 2022, pp. 1–20.

[51] J. Zhang, T. Wang, and X. Wan, *Pkusumsum: a java platform for multilingual document summarization*, in Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: System Demonstrations, 2016, pp. 287–291.

[52] Y. Zhao, X. Mao, and Q. Cui, *Domain-specific extractive and abstractive summarization for academic papers*, Journal of Computer Science and Technology, 34 (2019), pp. 770–783.

Edited by: Mudasir Mohd
Special issue on: Scalable Computing in Online and Blended Learning Environments: Challenges and Solutions
Received: Aug 29, 2023
Accepted: Sep 21, 2023