Journal of Big Data

RESEARCH Open Access

EXABSUM: a new text summarization approach for generating extractive and abstractive summaries

Zakariae Alami Merrouni[1*], Bouchra Frikh[1] and Brahim Ouhbi[2]

*Correspondence:
zakariae.alamimerrouni@usmba.ac.ma

[1] LIASSE Lab, National School of Applied Sciences (ENSA), Sidi Mohamed Ben Abdellah University, B.P. 72, Route d'imouzer, Fez, Morocco
[2] Mathematical Modeling and Computer Laboratory (LM2I), National Higher School of Arts and Crafts (ENSAM), Moulay Ismail University (UMI), Marjane II, B.P. 4024, Meknes, Morocco

Abstract

Due to the exponential growth of online information, the ability to efficiently extract the most informative content and target specific information without extensive reading is becoming increasingly valuable to readers. In this paper, we present 'EXABSUM,' a novel approach to Automatic Text Summarization (ATS), capable of generating the two primary types of summaries: extractive and abstractive. We propose two distinct approaches: (1) an extractive technique (EXABSUM$_{\text{Extractive}}$), which integrates statistical and semantic scoring methods to select and extract relevant, non-repetitive sentences from a text unit, and (2) an abstractive technique (EXABSUM$_{\text{Abstractive}}$), which employs a word graph approach (including compression and fusion stages) and re-ranking based on keyphrases to generate abstractive summaries using the source document as an input. In the evaluation conducted on multi-domain benchmarks, EXABSUM outperformed extractive summarization methods and demonstrated competitiveness against abstractive baselines.

Keywords: Extractive and abstractive summarization, Graph-based approach, Keyphrase-based approach

Introduction

The accessibility of the ever-expanding volume of online information by humans would be impeded without the presence of summaries. Given the extensive nature of textual content, pertinent information can inadvertently evade readers' attention. Consequently, the condensation of critical information into summaries holds significant value. Since the 1950s, researchers have diligently endeavored to enhance text summarization algorithms, with the aim of achieving a level of summarization comparable to human capabilities. Text summarization remains a formidable yet promising challenge within the domain of NLP.

In text summarization, two pivotal inquiries arise: (i) the process of identifying pertinent content within a document, and (ii) the art of succinctly conveying the selected material while minimizing redundancy [1–3]. The landscape of ATS approaches can be categorized into three primary categories: extractive, abstractive, and presently, emphasis is gravitating toward hybrid summarization—a fusion of extractive and abstractive techniques [4–6].

Despite the notable advancements in information technology, the domain of summarization remains an area necessitating substantial advancements. Within the realm of text summarization, several critical challenges persist, which can be encapsulated as follows:

- Initially, the challenge of Text Relevancy Detection emerges. Conventional methods assume that a word's significance within a text correlates with its frequency of occurrence, with each word representing a distinct concept. However, quantifying concept occurrences poses complexity due to the presence of synonymy and coreferential expressions that contribute to text cohesion. The information flow within a document exhibits fluctuations, indicating that specific segments hold greater importance than others. Consequently, the task of effectively discerning the most pertinent details and statically and semantically distinguishing relevant terms from source documents proves to be a pervasive challenge (e.g., selection predicated on pertinent keywords or keyphrases).
- Subsequently, the lack issue of coherence and redundancy. Extractive summarization faces hurdles of cohesion and coherence in the summaries produced, stemming from redundancy (phrases with comparable meaning), disjointed sentence connections, and unresolved co-reference relationships.
- The third challenge pertains to abstractive and hybrid summarization. The demand for abstractive or hybrid Automatic Text Summarization (ATS) techniques becomes apparent. This genre of technique remains an evolving and intricate domain. Crafting an efficacious abstractive summary has proven challenging thus far. It is imperative to cultivate overarching guidelines and viable strategies to transition from extractive to abstract summaries, thereby harnessing the advantages offered by both ATS approaches.

In this paper, we introduce EXABSUM, an ATS SYSTEM equipped to generate two distinct summary categories. Firstly, extracts ($\text{EXABSUM}_{\text{Extractive}}$) are shaped through a strictly extractive methodology, while abstracts ($\text{EXABSUM}_{\text{Abstractive}}$) are crafted via an abstractive approach. The outlined approach effectively addresses limitations intrinsic to both extractive and abstractive summarization techniques. Consequently, our contributions to state-of-the-art systems encompass the following:

- Diverging from certain extant extractive systems reliant solely on statistical scoring mechanisms for verbatim phrase extraction from the source document, our approach introduces a distinctive unsupervised extraction strategy aimed at tackling the challenge of Text Relevancy Detection. This innovative method combines the strengths of both statistical and semantic scoring techniques to discern crucial information, while concurrently proposing a novel one.
- Unlike certain extant extractive systems, our approach introduces the element of Semantic redundancy mitigation—a pivotal concern within ATS. To circumvent the inclusion of semantically and contextually redundant information in final

summaries, we advocate the adoption of textual entailment. This approach serves to mitigate the readability challenges inherent in existing methods, thereby alleviating a drawback commonly associated with the produced text.

- We confront the challenge of generating abstractive summaries by presenting a graph-based summarization model designed to yield resilient abstractive summaries. This model builds upon and extends a pioneering multi-sentence compression and fusion approach, bolstered by a re-ranking method based on key-extraction. Notably, this approach functions independently of any need for training data or acquiring knowledge of the document's structure or domain.

The paper's structure is delineated as follows. The subsequent section introduces pertinent related works and outlines ATS systems developed to cater to distinct applications. Sect. "EXABSUM ATS Approach" delves into the description of our proposed ATS system, EXABSUM. Within this section, we expound upon its primary stages, recommended architecture, and the two methodologies employed for the creation of extractive and abstractive summaries. In Sect. "Experimental setup", we detail the experimental framework. Here, we provide insight into the datasets utilized, elucidate the conducted experiments aimed at parameter tuning, and subsequently discuss the evaluation process. The achieved results, compared to the other state-of-the-art systems, are presented in the final part of the section. Finally, Sect. "Summary and conclusions" discusses the conclusion and future work.

Related works

The initial efforts in the domain of automatic summarization focused on extractive approaches, which aim to select pertinent existing words, phrases, or sentences directly from a source text to capture its most pivotal content. Extractive Automatic Text Summarization (ATS) approaches are typically carried out in three steps [5]: (1) Construct an intermediate representation of the original text (usually involving preprocessing and segmenting the text into paragraphs, phrases, and tokens); (2) Sentence scoring (the score should measure the importance of a sentence to the comprehensive understanding of the text) by attributing scores to the most relevant words, followed by an assessment of sentence characteristics such as position within the document, sentence length, title alignment, and other factors. Previous research of extractive summarization has predominantly focused on (1) sentence-clustering-based, (2) statistical, (3) graph-based, and (4) optimization-based techniques. In the context of the first approach, the document comprises n sentences, each sharing an identical set of terms. Consequently, the set of terms in the document corresponds to the set of terms in each phrase. The distance between corresponding sentences can be employed to illustrate the similarity in language patterns [7–10].

Sentence-clustering algorithms organize related textual units (paragraphs, sentences) into multiple clusters to uncover common themes of information, subsequently selecting text units from these clusters in the final summary. One of the noteworthy extractive summarization techniques is the centroid-based method [11]. An instance of an Automatic Text Summarization (ATS) system employing sentence-clustering algorithms is the MEAD system [12], a bilingual (English and Chinese) summarizer system that provides extractive single and multi-document generic or query-focused summaries. The MEAD system computes centroid topic characterizations for individual documents or provided clusters, leveraging tf–idf-type data. It evaluates candidate summary sentences by weighing sentence scores against the centroid, text position value, and tf–idf title/lead overlap. A summary length threshold governs sentence selection, while cosine similarity analysis against prior phrases curbs redundant new phrases.

Incorporating a summarization technique within a comprehensive retrieval and grouping process, the QCS system [13] generates a single extractive summary for each cluster. This is achieved through a method that combines sentence "trimming" and a hidden Markov model, followed by pivoting QR decomposition. The model identifies sentences with the highest likelihood for inclusion in the summary.

Statistical approaches [14] rely on elementary metrics like TF-IDF scores and word co-occurrence [1, 15, 16]. Ko and Seo [17] introduced a proficient methodology for text summarization that harnesses contextual insights and statistical methodologies to extract pertinent sentences.

Graph-based approaches [7] depict text as a network of phrases and devise summaries through graph-based scoring mechanisms. An innovative and versatile summarizer, GRAPHSUM, rooted in a graph model, was proposed by Baralis et al. [18]. It captures interrelationships among various elements by uncovering association rules. Parveen and Strube [19] presented an extractive graph-based unsupervised technique for summarizing individual documents that accounts for three critical summary attributes: significance, non-redundancy, and local coherence. Optimization-based methods [20] employ optimization techniques such as integer linear programming [21], constraint optimization [22], and sparse optimization [23].

Other ATS systems, like SummGraph [24], employ graph-based algorithms and knowledge databases to discern the substance of pertinent texts. Notably, this specific system has demonstrated efficacy across domains encompassing news, biomedical research, and tourism. Summaries have also embraced the incorporation of Natural Language Generation (NLG) to introduce fresh terminologies and linguistic structures. Belz [25] presents a text summarization technique grounded in 'NLG' to automatically generate weather forecast reports. Mohammad et al. [26] elucidated a system for the automated creation of technical surveys rooted in citations. More recently, Erera et al. [27] introduced the IBM Science Summarizer, an innovative methodology catering to Computer Science papers. This approach crafts summaries contingent upon user-provided information requisites, be it a natural language inquiry, scientific tasks (e.g., "Machine Translation"), datasets, or scholarly venues.

Although extractive methods can adeptly identify significant information, they may lack the fluidity and precision inherent in human-generated summaries. Consequently, abstractive ATS approaches strive to enhance sentence coherence by diminishing redundancies, elucidating sentence context, and potentially introducing supplementary phrases into the summary. For the synthesis of the final summary, abstractive techniques generally leverage sentence compression, fusion, or modification mechanisms. Barzilay and McKeown [28] pioneered a system wherein dependency trees represent input phrases, and select words are aligned to integrate these trees into a lattice structure. The lattice is subsequently linearized via tree traversal to generate fusion sentences.

Filippova and Strube [29] introduced an innovative approach to sentence fusion, framing the fusion task as an optimization problem. This unsupervised technique draws on dependency structure alignment, semantic and syntactically informed phrase aggregation, and pruning strategies. Later, Filippova delved into the challenge of condensing a collection of interconnected sentences into a succinct single sentence, termed as multi-sentence compression, and presented a foundational technique based on shortest paths in word graphs [30]. Her method yielded grammatically sound and informative summaries, subsequently finding application in diverse contemporary summary systems [4, 31]. Boudin [32] extended Filippova's approach by addressing Multi-Sentence Compression (MSC) as the task of generating a concise single-sentence summary from a cluster of interconnected sentences. He introduced an N-best reranking algorithm based on the frequency and relevance of keyphrases within the documents, resulting in more informative summaries. Banerjee et al. [33] devised multi-document abstractive summaries using word graphs and Integer Linear Programming (ILP). They clustered akin sentences among pivotal documents and employed word-graphs to identify shortest paths. The ILP model facilitated the identification of sentences with maximal information and readability, effectively reducing redundancy. Nayeem et al. [34] formulated an unsupervised abstractive summarization system. Their innovation was a paraphrastic sentence fusion model amalgamating sentence fusion with paraphrasing at the sentence level through a skip-gram word embedding model. This model augmented information coverage and heightened the abstract nature of the generated phrases. Shang et al. [35] introduced a fully unsupervised graph-based architecture tailored for abstractive summarization of meeting speeches. Their unified framework amalgamated the strengths of six prevailing approaches across three distinct tasks (keyword extraction, multi-sentence compression, and summarization), effectively addressing their respective limitations. Their abstractive summarization approach underwent four key processes: preprocessing, community recognition, multi-sentence compression, and submodular maximization.

Recently, the NLP research community has increasingly directed its attention towards Hybrid ATS techniques. In hybrid approaches, extractive methods are harnessed to identify content terms and sentences deemed essential for inclusion in the summary, while simultaneously guiding the development of abstracts [36]. Such methods amalgamate the strengths of both extractive and abstractive ATS techniques. Di Fabbrizio et al. [37] introduced a hybrid approach that crafts summaries for product and service reviews by blending natural language generation with salient sentence selection techniques. Their 'STARLET-H' system operates as a hybrid abstractive/extractive summarizer. It employs extractive summarization techniques to identify significant quotes from input reviews, incorporating them into an automatically generated abstractive summary to provide validation, disclosure, or justification for favorable and/or negative viewpoints. However, the algorithm necessitates a substantial amount of training data to comprehend aspect order. LLORET and ROM-FERRI [38] proposed the COMPENDIUM ATS system for generating research publication abstracts in the biomedical domain. This system produces two distinct types of generic summaries: extractive and abstractive-oriented, accompanied by their respective COMPENDIUM variants: COMPENDIUM-E and COMPENDIUM-A. The extractive approach selectively picks and extracts the most pertinent sentences, while the abstractive-oriented approach blends extractive and abstractive techniques, incorporating an information compression and fusion stage. Bhat et al. introduced "SumItUp," a single-document hybrid TS system, in [39]. The hybrid system consists of two phases: (1) Extractive Sentence Selection, which generates the summary using statistical features (sentence length, sentence position, TF-IDF, noun phrases, verb phrases, proper nouns, aggregate cosine similarity, and cue phrases), along with a semantic feature (emotion described in the text). In the extractive summary, cosine similarity is utilized to eliminate redundant sentences. For abstractive summary generation, the extracted sentences undergo processing by a language generator (a fusion of Wordnet, part-of-speech tagger, and Lesk algorithm) to transform the extractive summary into an abstractive rendition.

EXABSUM ATS approach

System's architecture

In this subsection, we explain the two approaches introduced by the EXABSUM ATS system for generating the two types of summaries. It is pertinent to highlight that our proposed ATS architecture comprises two distinct components. The first component, denoted as EXABSUM$_{\text{Extractive}}$, represents a purely extractive ATS approach (Sect. "EXABSUMExtractive core stages"), while the second component, EXABSUM$_{\text{Abstractive}}$, encompasses abstractive techniques to yield an abstractive summary (Sect. "EXABSUMAbstractive core stages").

EXABSUM$_{\text{Extractive}}$ core stages

The preliminary phase of our methodology is centered on extractive summarization. A conventional approach to extractive summarization treats sentences as individual entities, extracting the most pertinent ones from the text based on specific characteristic features (which gauge the suitability of a sentence for inclusion in the summary). Subsequently, the top N extracted sentences are organized to create the summary. The extraction procedure is compartmentalized into four stages (illustrated in Fig. 1).

The following core stages are covered in detail:

Text pre-processing

First, we initiate the process by conducting fundamental linguistic analysis to prime the text for subsequent stages of processing. This involves the application of text

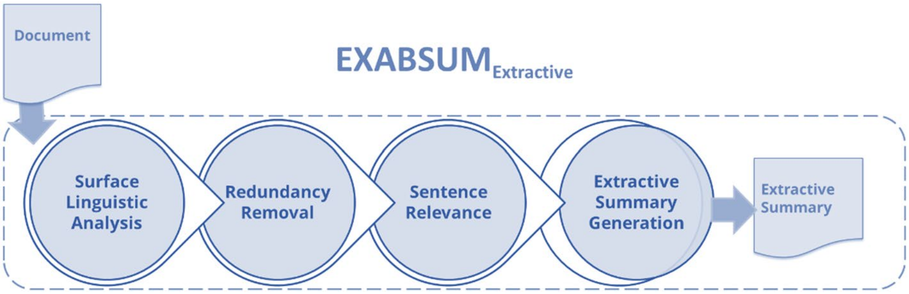

Fig. 1 EXABSUM$_{\text{Extractive}}$ stages for extractive summary. **a** preprocessing (surface linguistic analysis); **b** redundancy elimination; **c** sentence relevance; and **d** summary generation

pre-processing (TP) to standardize input files and establish clear sentence boundaries within word sequences. TP encompasses two primary categories: noise removal and normalization. Noise refers to data components that contribute redundancy to the primary text analytics. The manner in which this foundational phase is executed can significantly influence the accuracy of the sentence selection technique. Thus, it is imperative to provide explicit details regarding our implementation approach. Depending on the dataset type, each document undergoes the subsequent pre-processing stages:

- *Sentence splitting or segmentation*: As an initial step routinely conducted on texts prior to subsequent processing, this involves the process of dividing the input text into individual sentences. This division is undertaken to extract pertinent information from the text
- *Tokenization*: Each sentence undergoes intelligent tokenization, wherein all marks, punctuation, brackets, digits, and special characters are removed, and all words are converted to lowercase. For instance, given the sentence: "(text summurizagtnst Bion;,,;:,appR;aochAs is; NL = P a*nd I2r s)", the result would be: "(Text summarization approach is NLP and IR)". This process allows for the identification of individual words within the document, facilitating subsequent tasks such as calculating word co-occurrences and distinguishing between stop words and nonstop words.
- *Part-of-speech tagging*: Each word is assigned a morphological category using a part-of-speech tagger (such as noun, verb, adjective, preposition, adverb, determiner, pronoun, and conjunction). This process proves advantageous for discerning between various types of words, as certain categories (e.g., nouns or verbs) hold greater significance than others (e.g., determiners). This tool's application will be evident in subsequent data compression and fusion phases. Notably, the Stanford POS tagger was utilized for this part-of-speech tagging process.
- *Lemmatization*: Variations in a term can impact its frequency. Lemmatization involves reducing a word's inflectional forms and derivationally related forms to a standardized base form, referred to as its lemma. Unlike stemming, lemmatization relies on the precise identification of a word's intended part of speech and meaning within a phrase and in the broader context of surrounding sentences or even an entire document. To achieve this, we utilize the Stanford Core NLP package [40] to lemmatize our statements.
- *Stop Word Identification*: Certain stop words contribute to the reduction of feature space, resulting in decreased time and space complexity. Stop words encompass various prepositions, pronouns, and conjunctions commonly found in sentences. The removal of these terms prior to text analysis ensures that the prevalent words primarily pertain to the context rather than being commonplace throughout the text. In our process, this step is conducted before computing single keyword relevance, as stop words are excluded from consideration in subsequent phases.

Redundancy detection and removal

Redundancy is regarded as an undesirable attribute that affects the quality of summaries. In fact, the identified redundant sentences need to be removed from the texts, preserving only a single collection of non-repetitive sentences to be used as input for the summarization process. Our objective at this point is to identify semantically identical content within the source documents and exclude it from the summary. Textual Entailment (TE) is employed for this precise purpose [41].

The objective of TE is to determine whether the meaning of a text sample, referred to as a hypothesis (H), can be inferred from another text, known as the text (T) [41]. Textual Entailment (TE) involves predicting whether the information presented in the first sentence unquestionably implies the information in the second sentence for a pair of sentences. It addresses semantic inference as a direct mapping between linguistic expressions and abstracts the typical semantic inferences required for text-oriented NLP applications. TE has found successful application to the general summarization problem [42–44], and specifically for identifying duplicate information while addressing summarization [45]. The entailment relationships are computed using the TE method described in [46]. The TE tool relies on lexical (cosine similarity, Levenshtein distance), syntactic (dependency trees), and semantic measures based on WordNet [10].

After eliminating the redundant sentences from the source texts, the non-repetitive sentences that remain will be input into the extractive summarization approach. This approach employs a range of scoring techniques to identify pertinent content, encompassing both statistical and semantic aspects.

Sentence relevance

The significance of a sentence in relation to the overall comprehension of the text should be employed to ascertain its importance. This involves assigning scores to the most pertinent terms and subsequently assessing and computing sentence attributes such as document position, sentence length, and title similarity. These features can be integrated to assess the remaining sentences and select those with the highest scores for inclusion in the summary [47–51].

Sentence salience scoring techniques (or combinations thereof) are employed to assign a score to each sentence based on its significance. In this work, we introduce a hybrid model based on extraction, which integrates statistical, structural, and semantic features. The subsequent subsections offer a concise overview of the methods utilized in this phase:

a. Term Relevance-Inverse sentence frequency (TR-ISF)

We introduced a novel metric named TR-ISF, derived from the conventional Information Retrieval IR technique Term Frequency-Inverse Document Frequency (TF-IDF). This modified version of TF-IDF is tailored for sentence-level text summarization, as opposed to the document-level summary for which TF-IDF is traditionally used. In this approach, the relevancy TR of term t is established through its statistical and semantic relationship across the entire document-dataset level. Subsequently, the ISF gauges the descriptiveness of a word, assessing its prevalence or rarity across all sentences. This methodology operates under the assumption that if a term is both relevant and present in a limited number of sentences, it is likely to be included in the summary. In essence, pertinent keywords can be employed to detect or quantify sentence relevance, as well as to pinpoint the most relevant topic or topics within a text.

Initially, we employ a Hybrid Feature Selection Model (HFSM) to compute the term relevance using the 'TR' metric. This model integrates both statistical and semantic features. Subsequently, the TR-ISF Equation (Eq. (12)) is employed to ascertain the ultimate synthetic score for each term, which is subsequently leveraged to compute the sentence's salience score (Eq. (13)). It's important to note that not all terms are taken into account, and to ensure accuracy, stop word filtering and stemming are applied prior to evaluating a term's relevance.

The chi-square statistic permits the testing of statistical independence between a term and a category by contrasting the observed frequency with the expected frequency, calculated under the assumption of their independence. The χ^2 value is defined as:

$$\chi^2_{w,c} = \sum\nolimits_{i\in\{w,\overline{w}\}}\sum\nolimits_{j\in\{c,\overline{c}\}}\frac{\left(O\left(i,j\right)-E(i,j)\right)^2}{E(i,j)} \quad (1)$$

where $O\left(i,j\right)$ represents the observed frequency and $E\left(i,j\right)$ denotes the count of documents that fall under category c and also contain the term w. To discern the nature of the dependency when present, Li et al. [52] introduced a novel measure called term category dependency, defined as:

$$R_{w,c} = \frac{O(w,c)}{E(w,c)} \quad (2)$$

where $R_{w,c}$ is the ratio between $O(w,c)$ and $E(w,c)$. $R_{w,c}$ should be close to 1 if there is no dependency between the term w and the category c(i.e., $\chi^2_{w,c}$ is not statistically significant), $R_{w,c}$ should be larger than *1* if there is a positive dependency, meaning the observed frequency is greater than the expected frequency. Conversely, $R_{w,c}$ should be smaller than *1* if there is a negative dependency.

In order to calculate the feature significance of the word w within a corpus containing k categories, Li et al. [52] combine Eqs. (1) and (2) which results in a novel measure known as CHIR, defined as follows:

$$r\chi^2(w) = \sum_{j=1}^{k} p\left(R_{w,c_j}\right)\chi^2_{w,c_j}\ with\, R_{w,c_j} > 1 \quad (3)$$

where $p(R_{w,c_j})$ is the weight of chi-square statistic χ^2_{w,c_j} in the corpus in terms of R_{w,c_j}. It is defined as:

$$p\left(R_{w,c_j}\right) = \frac{R_{w,c_j}}{\sum_{j=1}^{k} R_{w,c_j}}\ with\, R_{w,c_j} > 1 \quad (4)$$

This new term-goodness measure, $r\chi^2(w)$, is the weighted sum of χ^2_{w,c_j} statistics when there is a positive dependency between the term χ^2_{w,c_j} and the category c_j, a bigger $r\chi^2(w)$ or CHIR measure value indicates that the term is more relevant.

We utilized the Mutual Information (SIM) measure, a commonly employed concept in information theory, to enhance the semantic aspect of the chosen words within a specific context. This measure quantifies the significance of words based on their semantic content and serves as a gauge of their value. SIM was introduced as a means of gauging word association, indicating the intensity of the connection between words by contrasting their actual probability of co-occurrence with the probability anticipated by chance.

Mutual Information indicates the proportionate shift in the likelihood of encountering x when y is present (the amount of information that y provides about x) [8]. It is based on the fact that two words are considered similar if their mutual information with all the words in the vocabulary V is nearly the same [8]. The semantic similarity measure between two terms w_1 and w_2 is defined as follows:

$$sim(w_1, w_2) = \frac{1}{2|V|} \sum_{i=1}^{|V|} \left(\frac{min(I(z_i,w_1),I(z_i,w_2))}{max(I(z_i,w_1),I(z_i,w_2))} + \frac{min(I(w_1,z_i),I(w_2,z_i))}{max(I(w_1,z_i),I(w_2,z_i))} \right) \tag{5}$$

where V is the vocabulary and $I(z_i, w_1)$ is the mutual information between the term z_i and w_1. $I(z_i, w_1)$ is evaluated using the following formula:

$$I(z_i, w_1) = P_d(z_i, w_1) log\left(\frac{P_d(z_i, w_1)}{P(z_i)P(w_1)}\right) \tag{6}$$

where d represent the size of a sliding window,$P_d(z_i, w_1)$ is the probability of succession of z_i and w_1 in a window of $(d + 1)$ words and $P(z_i)$ is the priori probability of the term z_i. This probability can be estimated by the ratio of the number of times that z_i is followed by w within the window and by the cardinal of the vocabulary.

The similarity between a term w and a document centroïd d is defined in [53] as the average of the similarities between the word w and the x words of the document centroïd. This measure is given by:

$$SIM(w, d) = \frac{\sum_{j=1}^{x} sim(w, w_j)}{\sum_{j=1}^{x} \sum_{i=1}^{x} sim(w_j, w_i)} \tag{7}$$

so as to determine the semantic relevance of a term w in a corpus of k clusters, for each cluster we calculate the weighted sum of its similarities with the document centroid $dcen_j$ of each cluster c_j using the following formula:

$$SIM(w) = \sum_{j=1}^{k} P\big(I\big(w, dcen_j\big)\big) SIM(w, dcen_j) \tag{8}$$

where $P\big(I\big(w, dcen_j\big)\big)$ is the weight of the similarity between the term w and the document centroïd $dcen_j$ and $I\big(w, dcen_j\big)$ is the mutual information between w and $dcen_j$. Considering the contingency table of a term w and a centroïd d where A is the number of times w and d co-occur i.e.,w occur in documents that belong to the cluster whose centroid is d, B is the number of times w occurs without d, C is the number of times d occurs without w and N is the total number of documents.

The mutual information criterion between a term w and a document $dcen_j$ is defined by:

$$I(w, dcen_j) = P(w, dcen_j)log\left(\frac{P(w, dcen_j)}{P(w)P(dcen_j)}\right) \tag{9}$$

If there is a strong association between w and $dcen_j$ then the joint probability $P(w, dcen_j)$ will be larger than $P(w)P(dcen_j)$; consequently $I\left(w, dcen_j\right) > 0$. If w and $dcen_j$ are in complementary distribution, then $P(w, dcen_j)$ will be less than $P(w)P(dcen_j)$ hence $I(w, dcen_j) < 0$. In the case of poor association between w and $dcen_j$, then $P(w, dcen_j) \approx P(w)\ P\ (dcen_j)$, consequently $I(w, dcen_j) \approx 0$. The weight of $P\left(I\left(w, dcen_j\right)\right)$ defined as*:*

$$P\left(I\left(w, dcen_j\right)\right) = \frac{I\left(w, dcen_j\right)}{\sum_{i=1}^{k}\left(I\left(w, dcen_j\right)\right)} withI\left(w, dcen_j\right) > 0 \tag{10}$$

A term with a high weight in the SIM(w) metric implies that it is semantically relevant.

We define the feature goodness of a term as a combination of its statistical measure $chir(w)$, and its semantic measure$sim(w)$. The overall measure of a term's relevance,$TR(w)$, is defined as follows:

$$TR(w) = \alpha * chir(w) + (1 - \alpha) * sim(w) \tag{11}$$

where α is a weighting parameter between 0 and 1.

To select the most p pertinent terms, three steps are followed: (1) calculate the hybrid measure $TR(w)$ for each term in the document and the dataset, (2) sort the term in descending order of their criterion function, and (3) finally select the top p terms from the sorted list. A threshold δ is set to 0.25 to filter terms with a low $TR(w)$ value. In other words, the higher the relevancy of a word, the more important it is in indicating the main topic of a document.

Hence, the $\boldsymbol{TR - ISF}$ of a word is computed as shown in Eq. (12) and the salience score of a sentence is calculated as presented in Eq. (13).

$$TR - ISF(w_i) = TR(w_i) \times \log\left(\frac{S}{S_{w_i}}\right) \tag{12}$$

$$TR - ISF(s_i) = \sum_{w_i \in T}^{S} TR - ISF(w_i) \tag{13}$$

where

- TR returns the relevancy of a term(word) w_i in the document(s),
- T is the total of terms (words) in s_i,
- S_{w_i} is the total of sentences in which a relevant word w_i is presented (calculated by Eq. 11),
- S is the total of sentences in the document.

b. Sentence resemblance to the title

The title of a document often captures the main subjects discussed within it, particularly in news articles and scientific publications. The "sentence resemblance to the title" methodology assesses the similarity between sentences in a document and its title. By employing this technique, we deduce that sentences exhibiting greater similarity to the title signify the primary topic addressed in the document. This feature is computed as illustrated in the following Equation:

$$SenRT(S_i) = \frac{w_{s_i} \cap w_t}{|w_t|} \tag{14}$$

where,

- w_{s_i} is the set of the relevant words in s_i
- w_t is the set of words in the title,
- $|w_t|$ is the total of words in the title.

c. Sentence length

The consideration of sentence length aims to avoid selecting sentences that might be too short to convey the document's key points, as well as sentences that are excessively long and may result in wasted space. Acknowledging the possibility that a sentence could contain essential information in one part and unrelated information in another, this method takes into account the sentence's word count as a measure of its length.

This approach is utilized to discourage the selection of sentences that are either excessively short or excessively long, as they are not deemed optimal. Initially, sentences that fall below a specific size threshold (sentences with fewer than ten non-stop words) or exceed a certain length (sentences containing more than 50 non-stop words) are filtered out before computing the sentence score. Subsequently, the remaining sentences are assigned scores as depicted in Eq. (15).

$$SentenceLen(s_i) = \frac{\#number\_of\_words\_in\_s_i}{\#\max\_number\_of\_words\_in\_a\_sentence} \tag{15}$$

In practice, the penalty score is determined by a conditional:

$$Score(S_i) = \begin{cases} L_i & \text{if}(\mathrm{L_i} > C) \\ L_i - C & \text{othetwise} \end{cases} \tag{16}$$

where,

- L_i is the length of sentence i and
- C is a certain length defined by user.

d. Sentence position

The sentence position heuristic is among the most effective strategies for selecting relevant sentences in automatic text summarization (ATS). This heuristic operates on the assumption that the introductory sentences within a document hold the most crucial information. As the document unfolds, the significance of sentences tends to diminish. In our approach, we prioritize sentences that are located closer to the beginning of a document.

The score for this feature is calculated using the following formula:

$$SentPosition(S_i) = 1 - \frac{i}{S} \tag{17}$$

where

- i is the i_{th} sentence in the document, with i starting by zero,
- S is the total of sentences in the document.

Summary generation

Once the scores for each sentence have been computed, the objective of this stage is to create a summary by arranging sentences based on their relevance scores. The highest-scoring sentences are selected and extracted in the order they appear in the original document, resulting in a meaningful extractive summary. To determine the overall significance of a sentence, we employed the averaged combination approach, which is considered the most effective combination method and often leads to substantial improvements [48, 54]. The salience score of a sentence is determined by averaging the individual scores obtained through the N considered scoring procedures in this combination.

EXABSUM$_{\text{Abstractive}}$ core stages

This stage aims to create an abstractive summary through the generation of new text that captures the core content or conceptual elements of the original text. This summary succinctly and coherently communicates the primary information within the document. For this purpose, we employ a graph-based approach to construct a comprehensive

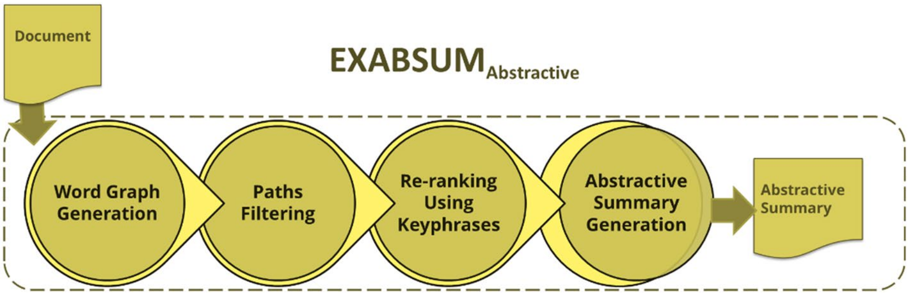

Fig. 2 EXABSUM$_{\text{Abstractive}}$ stages for generating abstractive summary

abstractive summary, followed by a re-ranking stage that relies on keyphrases. The following steps (Fig. 2) outline the main procedures involved in this stage:

Word graph generation and re-ranking

To generate a summary containing novel sentences, this stage involves compressing and merging sentences, followed by a re-ranking process based on the quantity and relevance of keyphrases present. This approach has demonstrated its efficacy in producing more informative summaries [30, 32].

A weighted directed word graph is constructed using a document (represented as a directed weighted graph) as input. Nodes in the graph correspond to words, and edges signify adjacency relationships between pairs of words. Each edge's weight is determined by the reciprocal frequency of co-occurrence of the two words.

Once the document is transformed into a word graph, a set of new sentences is generated by identifying the shortest path between nodes. This begins with the first word of each phrase in the extracted document, spanning its entire content. The following details the methodology:

Let G = (V,E) be a directed graph with vertices (nodes) V and directed edges E, where E is a subset of V*V. Given a set of related sentences $S = (s_1, s_2, \ldots, s_n)$, a word graph is constructed by iteratively adding sentences to it.

Figure 3 illustrates the word graph built from the four provided sentences. Edge weights have been omitted for clarity, and italicized sentence fragments are represented by dots.

1. The president of U.S. Donald Trump visited Venezuela last Thursday.
2. Donald Trump did a visit to the People Republic of Venezuela on Thursday.
3. Last week the President of State M. Trump visited Venezuela officials.
4. Donald Trump wanted to visit Venezuela last month but suspended his arrangements till Thursday last week.

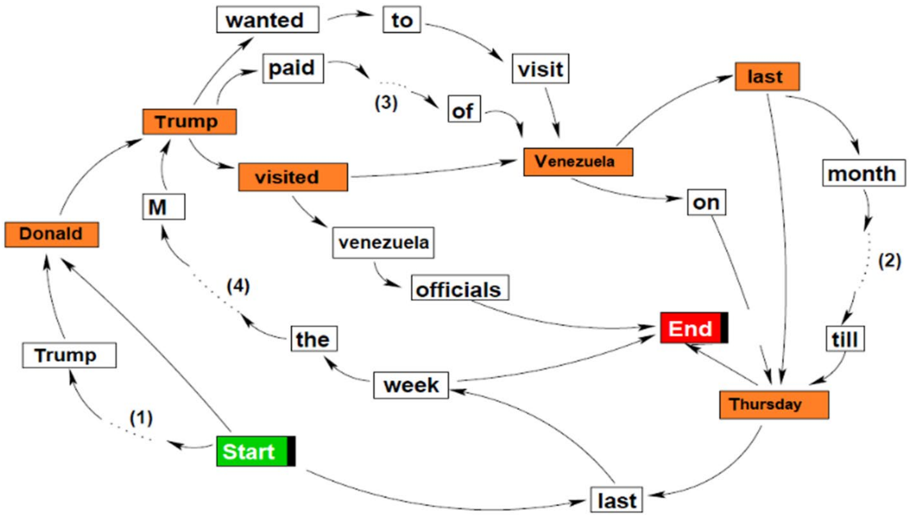

Fig. 3 Word graph constructed from sentences (1–4), along with a potential compression path

In the first step, the graph represents a single sentence (a sequence of word nodes without punctuation) along with the start and end symbols (depicted as start and end symbols in Fig. 3). For each word in the sentence, a corresponding node is added to the graph, and directed edges connect words that are adjacent in the sentence. If two words in subsequent sentences share the same lowercase form, they are linked to an existing node in the graph, provided that no word from the current sentence has been associated with that node before. Incorporating part-of-speech (POS) information reduces the likelihood of combining verbs with nouns (e.g., "visit"), thus preventing the generation of ungrammatical sequences. In cases where no suitable candidates exist in the graph, a new node is generated.

The process of word mapping and creation (adding words to the graph) is carried out in three distinct steps during the second stage:

1. non-stop words for which no candidate exists in the graph or for which an unambiguous mapping is possible;
2. non-stop words for which there are either several possible candidates in the graph, or which occur more than once in the sentence
3. stop words

For the last two groups of words where the mapping is ambiguous (i.e., there are two or more nodes in the graph that refer to the same word / POS tuple), the immediate context (the preceding and following words in the sentence and the adjacent nodes in the graph) is examined. As a result, the candidate that exhibits a greater overlap in context is selected. Alternatively, the candidate node with the highest frequency (i.e., the node with the most words mapped to it) is chosen. In Fig. 3, for example, when sentence (3) is to be inserted, there are two potential candidate nodes for "last". Stop words are only linked if they overlap with their non-stop word neighbors. If this condition is not met, a new node is created. We utilize the NLTK stop word list, supplemented with temporal nouns (e.g., Thursday, today). Filippova's method prohibits the inclusion of punctuation marks. Boudin and Morin [32] introduced a fourth step for constructing well-punctuated compressions, involving the addition of punctuation marks to the graph. When ambiguity arises in mapping, the candidate with the same immediate context is preferred. Words contiguous in the sentence are connected with directed edges once the sentence's words are added to the graph.

The weighting function is defined in Eq. 18 to compute edge weights and determine the optimal path, representing the most effective compression for the input sentences.

$$w\left(i,j\right) = \frac{cohesion(i,j)}{freq(i) \times freq(j)} \tag{18}$$

$$cohesion\left(i,j\right) = \frac{freq(i) + freq\left(j\right)}{\sum_{s \in S} d(s,i,j)^{-1}} \tag{19}$$

where $freq(i)$ is the number of words mapped to the node i. The function $d(s, i, j)$ refers to the distance between the offset positions of words i and j in sentence s.

This function has two objectives:

(1) to achieve grammatical compression, it prioritizes connections between words that frequently appear in a particular order (refer to Eq. 19).
(2) to generate an informative compression, it promotes paths passing through salient nodes.

The weighting function utilized in the K-shortest path algorithm serves to identify the shortest paths within the graph from the starting point to the endpoint (Eq. 20). Paths with a length of less than eight words or those lacking a verb are filtered out. The remaining paths are subsequently re-evaluated by normalizing the cumulative weight of the path over its length. Consequently, the path with the lowest average edge weight is considered the optimal compression. In our scenario, the initial node corresponds to the first word of each sentence during the generation of new sentences. This ensures that every sentence in the source text yields at least one derived sentence, guaranteeing comprehensive coverage of the document's content.

Paths filtering

Following the compilation of sentences through the shortest pathways, it's possible that certain sentences are nonsensical, improperly constructed, or incomplete. Therefore, a filtering stage is imperative to discard inappropriate pathways and uphold the integrity and coherence of the statements. To achieve this, we establish rules that necessitate sentences to satisfy all of the defined criteria; those that fail to do so are disregarded.

These rules are defined as follows:

- Every sentence must contain a verb.
- A sentence must be at least three words long.
- The sentence should not end in an article (e.g., a, the), a preposition (e.g., of), an interrogative word (e.g., who), or conjunction (e.g., and).

Upon the removal of erroneous sentences, the replacement sentences can seamlessly substitute the original ones.

Re-Ranking candidate sentences using keyphrases

Despite the apparent effectiveness of Filippova's method, a notable drawback is the absence of substantial information in a range of 48 to 60% of the generated sentences [30]. This limitation arises because node salience is solely determined by the frequency measure. In response to this concern, we proposed a re-ranking approach that re-evaluates the N-best list of compressions by considering both the quantity and significance of keyphrases present within them. A truly informative and pertinent sentence is expected to incorporate the most relevant keyphrases [55].

Hence, we integrated a re-ranking stage that prioritizes compressions featuring the most pertinent keyphrases derived from the initial set of input sentences. This additional step involves re-evaluating the N-best multi-sentence compression candidates generated through the word graph-based method, considering the quantity and importance of keyphrases encompassed within each candidate compression.

We opted for the shortest path approach followed by a re-ranking step due to three main reasons:

1. Retaining Salient Terms: the shortest path method allows us to compress sentences while retaining important terms from the original input. It also facilitates grouping words that frequently appear together in many sentences.
2. Inclusion of Content: by fusing multiple sentences, we can incorporate more content into the summary, enhancing its comprehensiveness.
3. Improved Informativeness: the re-ranking stage further enhances the summary by maximizing the diversity of covered topics and producing informative and grammatically accurate sentences. The utilization of keyphrases aids in crafting sentences that effectively capture the core ideas across a set of interconnected statements.

The unsupervised technique by Wan and Xiao [56] involves extracting significant words from interconnected sentence groups. This approach is built on the concept that a word's importance can suggest the presence of other words that often occur together. The strength of this suggestion is recursively determined based on the significance of the suggesting word.

To initiate the process of keyphrase extraction, a weighted graph is constructed from the connected sentences. In this graph, nodes represent words, identified as word and POS tuples. When two words co-occur in a sentence, corresponding nodes are connected by edges, with edge weights denoting the frequency of their co-occurrence. The TextRank algorithm [57], a graph-based ranking method that incorporates edge weights, is employed to compute the salience score for each node. The score for a node V_i is initialized with a default value and is iteratively calculated until it converges using the following Equation:

$$(V_i) = \frac{1-d}{\mathrm{N}} + d \times \sum_{V_j \in adj(V_i)} \frac{W_{ji}}{\sum_{V_k \in adj(V_i)} w_{jk}} S(V_i) \tag{20}$$

where $adj(V_i)$ represents the neighbors of V_i and d is the damping factor set to 0.85.

The second phase involves generating and evaluating potential keywords. We merge sequences of adjacent words that adhere to a given syntactic pattern to create multi-word phrases. In our case, we defined noun phrases based on our POS tag definitions, satisfying the regular expression rule: (NN | NNS | NNP | NNPS | VBN | JJ | JJS | RB) * (NN | NNS | NNP | NNPS | VBG) + . Unlike other definitions, our noun phrase structure includes adverbial nouns (tag RB) like "double experience" (RB NN) and present participle verbs (tag VBG) such as "virtual desktop conferencing" (JJ NN VBG), with the VBG tag appearing at various positions within the noun phrase. Adverbial nouns, also known as adverbial objectives, occupy the position that a verb's direct object typically occupies and modify the verb by providing information about time, distance, weight, age, or monetary value. Adverbs can interact with noun phrases, impacting the context and meaning of a candidate keyphrase. This interaction is particularly notable in scientific contexts, where authors are precise in explaining specific situations.

The score of a candidate keyphrase k is calculated by summing the salience scores of the words it contains, normalized by its length + 1 to favor longer n-grams (as shown in Eq. 21).

$$\text{score}(k) = \frac{\sum_{w \in k} \text{TextRank}(w)}{\text{length}(k) + 1} \tag{21}$$

The generated keyphrases are grouped into clusters based on word overlap. In each cluster, the keyphrase with the highest score is selected. This filtering process produces a smaller subset of keyphrases that better represent the content of the cluster. However, the limited scope of the N-best list can hinder the effectiveness of re-ranking techniques, as they may discard many potentially suitable candidates. To address this, various other paths are considered. These paths are re-ranked by normalizing the overall weight of the path (as defined in Eq. 18) across its length and then multiplying it by the sum of the key scores it contains. The score for sentence compression c is determined as follows:

$$\text{score}(c) = \frac{\sum_{i,j \in \text{path}(c)} w_{(i,j)}}{\text{length}(c) \times \sum_{k \in c} \text{score}(k)} \tag{23}$$

Abstractive summary generation

The objective of this concluding stage is to create an abstractive summary based on the input document. Once the preceding processes have been carried out, the remaining sentences are employed to generate abstractive summaries. Through these stages, an abstractive summary is produced, composed of properly structured and complete sentences extracted using the shortest paths. Among these, the top N relevant sentences are selected, considering their high number of keyphrases. Consequently, the resulting summaries encompass abstractive content. These summaries are categorized as abstracts, as they do not replicate the exact sentences found in the source document.

Experimental setup

A comprehensive evaluation of EXABSUM's performance has been conducted using diverse corpora spanning a wide array of topics. In this section, we will outline the following aspects: (i) the datasets employed and the methodologies used to assess the experiments; (ii) the experimentation process involving parameter refinement. Ultimately, we will compare our results with those of other analogous works.

Datasets

It is common practice to assess an algorithm by conducting experiments on a specific corpus of text summarization tasks, which encompasses both the source texts and manually generated summaries. In our case, we employed several datasets from diverse domains as our corpora. By encompassing domains like newswire, tourism, Web 2.0, science, business, health, justice, lifestyle, opinion, politics, entertainment, sports, technology, and travel, EXABSUM's evaluation is carried out from a comprehensive perspective.

EXABSUM's evaluation focuses on the following datasets:

- DUC 2001 and DUC 2002: these datasets are widely used in ATS tasks and were provided by the Document Understanding Collection (DUC) and Text Analytics Conferences (TAC). DUC 2001 consists of 309 English news articles, each accompanied by two separate golden summaries prepared by different individuals. DUC 2002 contains 567 news articles in English, covering various topics and lengths, and also includes two gold-standard summaries. The length of the accompanying summaries for both datasets is approximately 100 words. Notably, the DUC collections are sentence-divided to identify the most informative sentences. The DUC datasets are organized into different categories, including biography, politics, law, society, culture, business, health, natural disasters, science, sports, and international topics. Certain categories like 'Natural Disaster,' 'Politics and History,' and 'Natural Disaster' constitute a significant portion of DUC 2002, making up about 60% of the documents in these categories. All DUC publications and clusters include human-generated summaries of approximately 100 words.
- *CNN Corpus* [58] is a substantial collection of news documents used for single-document summarization tasks, sourced from CNN's website (http://www.cnn.com). This corpus stands as the largest available dataset for single-document extractive summarization. It comprises 3,000 English articles that are grouped into twelve subject categories, as originally categorized by CNN: Business, Opinion, Politics, Showbiz, Health, Justice, Living, Sports, World News, Technology, United States, and Travel. The CNN Corpus offers high-quality abstractive summaries for each document, known as "highlights," which are authored by the original writers. In addition to these abstractive summaries, extractive summaries (gold standards) are also provided, each containing around 90 to 100 words. These summaries serve as essential references for both qualitative and quantitative assessments of automated summarization methods. The gold standard summaries encompass approximately 10,754 sentences, constituting around 10% of the total number of sentences in the 3,000 texts of the CNN Corpus. Numerous research projects are employing the CNN Corpus, ranging from addressing dangling co-references to enhancing extractive summarization techniques and even generating abstractive summaries from extractive ones. Notably, the CNN Corpus was used in the DocEng'19 Extractive Text Summarization Competition [58, 59]. This rich dataset plays a crucial role in advancing the field of automatic summarization.

Table 1 offers an overview of the datasets utilized in this study, providing basic information about each corpus. The table showcases details such as the number of clusters, document domains, total document count in each dataset, total sentence count, available test documents, average summary length in terms of words, and the intended task for each corpus.

Table 1 Statistics of the CNN and DUC datasets

Dataset	Number of clusters	Domain's documents	Number of Documents	Sentences	Number of test documents	Avg. length (Model sum)	Task
DUC01	30	Multi-Domain	309	269,990	309	100	Single and Multi
DUC02	59	Multi-Domain	567	348,012	567	100	Single and Multi
CNN	0	Multi-Domain	3000	2,628,336	2000	90	Single

Evaluation method

We conducted two types of evaluations: quantitative and qualitative. In the quantitative evaluation, we employed state-of-the-art assessment methods to compare our outcomes with the gold-standard models of the articles. The qualitative evaluation aimed to determine the extent to which our generated summaries comprehensively covered the key topics of the research articles. Thus, we evaluated the summaries in terms of user satisfaction.

Quantitative evaluation

In the quantitative evaluation, we measure the similarity between a set of candidate summaries and a collection of reference models (gold standard summaries). This evaluation aims to assess the informativeness of the summaries in terms of their content. To achieve this, we utilize the ROUGE-N metric, which captures various levels of N-gram co-occurrences between candidate summaries and reference models. Notably, ROUGE-1 and ROUGE-2 are well-known ROUGE metrics that compute the overlaps of unigrams and bigrams. Among these metrics, ROUGE-1 recall exhibits the strongest recall ability to identify a better summary within a pair [60, 61]. ROUGE-N quantifies the n-gram recall between a candidate summary and a set of reference summaries using the following formula:

$$ROUGE-N=\frac{\sum_{S\in\{\mathrm{ReferemceSummaries}\}}\sum_{\mathrm{gram}_n\in S}\mathrm{Count}_{\mathrm{match}}(\mathrm{gram}_n)}{\sum_{S\in\{\mathrm{ReferenceSummaries}}\sum_{\mathrm{gram}_n\in S}\mathrm{Count}(\mathrm{gram}_n)} \tag{24}$$

where n stands for the length of the n-gram, gram_n, and $\mathrm{Count}_{\mathrm{match}}(\mathrm{gram}_n)$ is the maximum number of n-grams co-occurring in a candidate summary and a set of reference summaries [60, 61]. Lin [60, 61] also demonstrated a strong correlation between ROUGE-1 recall and human judgments. Additionally, we employ ROUGE-SU4, which counts overlapping skip-bigrams between a candidate summary and a reference model, allowing for a maximum gap of four words. Lastly, we use ROUGE-L, which measures the longest common subsequence between two summaries [60, 61].

Qualitative evaluation

In this evaluation, our objective is to measure user satisfaction with the generated summaries. For this purpose, we carried out a qualitative evaluation by inviting ten English-speaking individuals to rate our summaries. We adopted the same qualitative evaluation method outlined in [38]. To illustrate, while a 3-level scale might include the categories "low," "medium," and "high," a 5-level Likert scale provides varying degrees to gauge agreement on a specific matter, ranging from "strongly agree," "agree," "neither agree nor disagree," "disagree," to "strongly disagree."

Specifically, the asked questions are:

- Q1: The summary reflects the most important issues of the document.
- Q2: The summary allows the reader to know what the article is about.
- Q3: After reading the original summary provided with the document, the alternative summary is also valid.

Given the diverse lengths of the documents, our evaluation approach focused on utilizing 10 randomly selected documents from each of the tested datasets.

Experiments results and discussion

In this section, we conducted experiments to evaluate different types of EXABSUM summaries. We employed two variations of EXABSUM: (i) $\text{EXABSUM}_{\text{Extractive}}$, which generates extractive summaries, and (ii) $\text{EXABSUM}_{\text{Abstractive}}$, which generates abstractive summaries. By evaluating both types of summaries, we aimed to assess EXABSUM's ability to extract relevant information and its performance in addressing the abstractive text summarization challenge. Additionally, we aimed to determine whether the strategies employed in EXABSUM are effective in generating summaries across various domains, such as Business, Opinion, Politics, Showbiz, Health, Justice, Living, Sports, Technology, Travel, newswire, and more. We compared the results with those of existing automatic text summarization systems to strengthen our findings.

Parameter value selection and analysis of scoring techniques' suitability

The aim of this evaluation is to appraise the effectiveness of the proposed features for sentence relevance detection. This assessment involves examining these features both individually and in combination, as outlined in the relevant subsections.

The weighting parameter α, as specified in Eq. (13), plays a crucial role in determining the relevance of a specific term within a sentence. This parameter governs the weight assigned to both the statistical feature (CHIR) and the semantic feature (SIM) within the hybrid weighting model. To evaluate the impact of each feature on the keyword's relevancy measure TR(w), we conducted multiple trials using different α values ($\alpha \in \{0, 0.2, 0.5, 0.6, 0.8, 1\}$). Our findings show that the most favorable outcomes are achieved when $\alpha = 0.6$ is utilized, closely followed by $\alpha = 0.5$. This observation underscores the significance of combining both statistical and semantic relationships to enhance the overall relevancy determination.

Table 2 ROUGE results for EXABSUM$_{\text{Extractive}}$ and EXABSUM$_{\text{Abstractive}}$ in Feature Analysis of the DUC2001 Dataset: Comb Represents the Combination of Selected Scoring Approaches

Summary type	*Scoring techniques*	**DUC 2001**	
		ROUGE-1	***ROUGE-2***
EXABSUM$_{\text{Extractive}}$	$TR - ISF$ ONLY	0.398	0.131
EXABSUM$_{\text{Extractive}}$	Comb1: $TR - ISF$+ Sentence position	0.402	0.143
EXABSUM$_{\text{Extractive}}$	Comb2: $TR - ISF$+ Sentence Position + Sentence Length	0.453	0.192
EXABSUM$_{\text{Extractive}}$	Comb3: $TR - ISF$+ Sentence Position + Sentence Length + Sentence resemblance To the Title	0.480	0.208
EXABSUM$_{\text{Abstractive}}$	Graphs + Reranking based on keyphrases ONLY	0.319	0.151

Table 3 ROUGE results for EXABSUMExtractive on DUC2002 dataset: analysis of features with comb as the combined scoring techniques

Summary Type	**Scoring Techniques**	**DUC 2002**			
		ROUGE-1	***ROUGE-2***	***ROUGE-SU4***	***ROUGE-L***
EXABSUM$_{\text{Extractive}}$	$TR - ISF$ ONLY	0.439	0.188	0.205	0.396
EXABSUM$_{\text{Extractive}}$	Comb1:$TR - ISF$+ Sentence position	0.441	0.189	0.207	0.401
EXABSUM$_{\text{Extractive}}$	Comb2: $TR - ISF$+ Sentence Position + Sentence Length	0.488	0.236	0.251	0.442
EXABSUM$_{\text{Extractive}}$	Comb3: $TR - ISF$+ Sentence Position + Sentence Length + Sentence resemblance To the Title	0.493	0.257	0.288	0.472
EXABSUM$_{\text{Abstractive}}$	Graphs + Reranking based on keyphrases ONLY	0.341	0.104	0.134	0.322

In our experiments, we consistently set the TR-ISF measure parameter α to 0.6. To comprehensively assess the effectiveness of various sentence relevancy scoring techniques, we conducted an ablation study using a backward-like total exclusion procedure. This involved individually excluding or adding the scores from each approach in the weighted averaged model. This evaluation enabled us to achieve three objectives: (1) determining whether the scoring techniques are suitable for enhancing ROUGE scores; (2) identifying their contribution to topic coverage within the document; and (3) gauging the sentence importance.

The ablation technique allowed us to compute ROUGE-1 and ROUGE-2 scores for the DUC01 dataset (Table 2), as well as ROUGE-1, ROUGE-2, ROUGE-SU4, and ROUGE-L scores for the DUC02 dataset (Table 3). Additionally, we presented the ROUGE-1 and ROUGE-2 scores for the CNN dataset in Table 4. Visual representations of the performance of our proposed approach under varied scoring technique values are depicted in Figs. 4, 5, and 6 for the DUC01, DUC02, and CNN datasets, respectively.

In the first experiment, as depicted in Tables 2 and 3, we focused on selecting summary sentences that include relevant keywords and topics determined by the TR-ISF scoring technique, corresponding to the sentence relevance identification stage of EXABSUM$_{\text{Extractive}}$. The TR component pinpoints significant keywords that signify essential topics, while the ISF component gauges a word's descriptiveness. We then compared the resulting combinations to the output of EXABSUM$_{\text{Abstractive}}$. In generating the

Table 4 ROUGE results for EXABSUM$_{Extractive}$ and EXABSUM$_{Abstractive}$ on CNN dataset while analyzing their features, Comb denotes the combination of the selected scoring techniques

Summary type	Scoring techniques	CNN	
		ROUGE-1	*ROUGE-2*
EXABSUM$_{Extractive}$	*TR* − *ISF* ONLY	0.519	0.351
EXABSUM$_{Extractive}$	*Comb1*: *TR* − *ISF* + Sentence position	0.541	0.379
EXABSUM$_{Extractive}$	*Comb2*: *TR* − *ISF* + Sentence Position + Sentence Length	0.592	0.434
EXABSUM$_{Extractive}$	*Comb3:TR* − *ISF* + Sentence Position + Sentence Length + Sentence resemblance To the Title	0.601	0.451
EXABSUM$_{Abstractive}$	Graphs + Reranking based on keyphrases	0.501	0.332

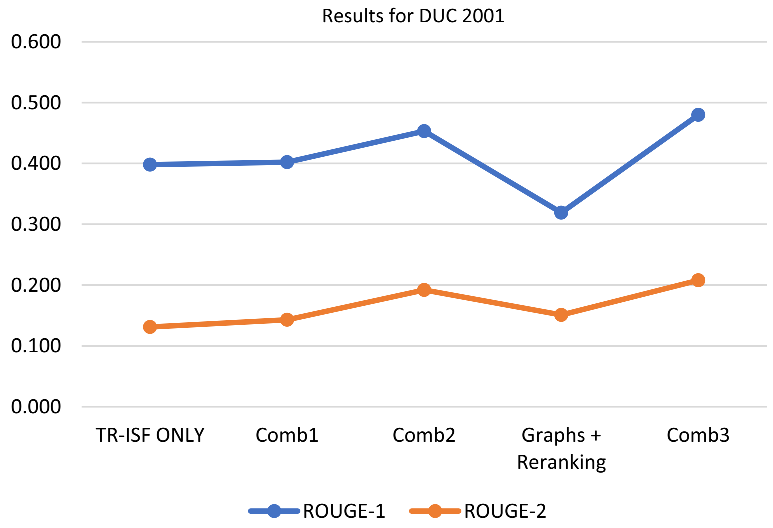

Fig. 4 ROUGE-1 and ROUGE-2 Results of EXABSUM on the DUC 2001 Collection with Varied Scoring Techniques and Summary Types

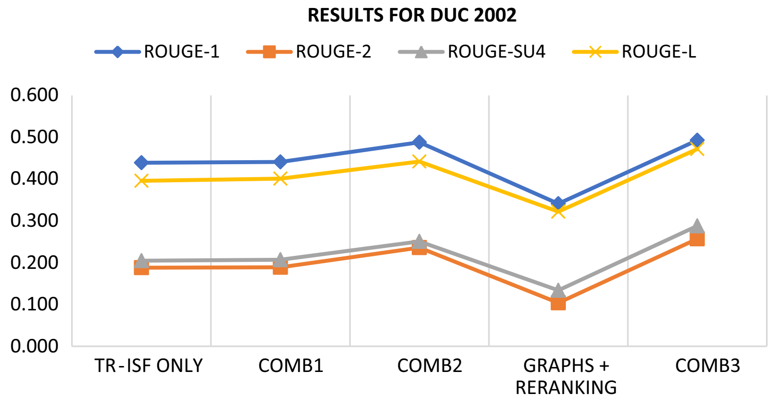

Fig. 5 ROUGE-1, ROUGE-2, ROUGE-SU4, ROUGE-L Results of EXABSUMExtractive on the DUC 2002 collection: variations in scoring techniques and summary types

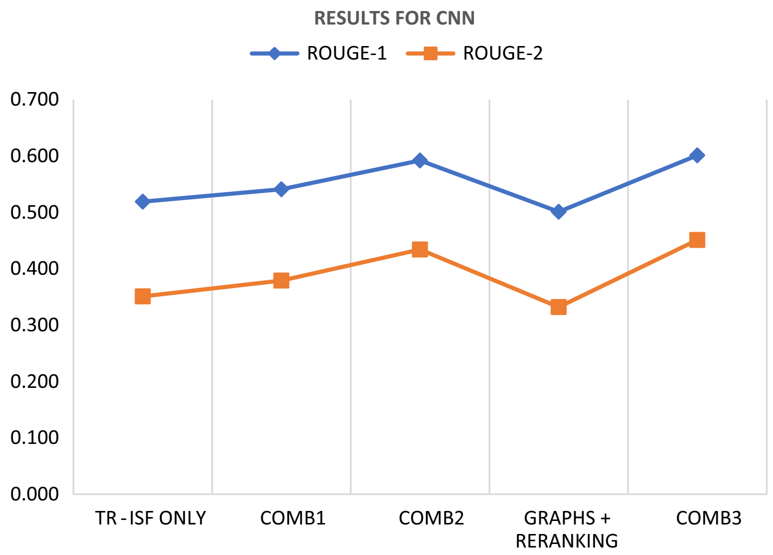

Fig. 6 ROUGE-1 and ROUGE-2 results of our proposed approach for the large CNN dataset while varying the scoring techniques and summary types

summary, we assigned scores to phrases using Eq. 13, and the highest-ranked phrases were utilized to construct the summary.

In the second experiment, we focus on selecting summary sentences based solely on sentence resemblance to the title, sentence length, sentence position, or a combination of these factors. As evident from Tables 2, 3, and 4, a clear trend is observed in most cases with $\text{EXABSUM}_{\text{Extractive}}$ yielding the best results, particularly when all scoring methodologies are combined (comb3). For instance, the ROUGE-1 results for $\text{EXABSUM}_{\text{Extractive}}$ with combination 3 show an average improvement of 13.44% compared to the $\text{EXABSUM}_{\text{Extractive}}$ approach that solely employs the TR-ISF for scoring phrases.

In the case of Combination 2, the same approach yielded an improvement of 11.85% over the results obtained for EXABSUMExtractive using only TR-ISF.

In terms of individual feature analysis, it is noteworthy that summaries generated solely utilizing the TR-ISF scoring technique generally perform well. This could be attributed to the use of a robust scoring technique (incorporating statistical and semantic features) to identify the most relevant terms or topics in a document. However, the results show improvement when the three recommended features are combined in the same approach (Comb3). Consequently, the well-incorporated features are well-suited for the extractive text summarization task, especially in the case of $\text{EXABSUM}_{\text{Extractive}}$. The superior ROUGE scores achieved by our system stem not only from the incorporation of TR-ISF and other scoring methodologies but also from the inclusion of the redundancy elimination phase using The Textual Entailment (TE) tool [62]. This phase plays a crucial role in generating semantically and syntactically non-redundant summaries. It identifies and removes sentences that are semantically redundant within documents. As a result, sentences with contextual overlap in other sentences can be omitted, leading to improved precision scores and overall system performance.

Regarding EXABSUM$_{\text{Abstractive}}$, based on preliminary findings, it is evident that relying solely on graphs and re-ranking based on key approaches does not yield high ROUGE scores, although the results are promising for future research endeavors. The moderate performance of this abstractive technique can be attributed to the constrained summary length of 100 words. Consequently, the selection process for the most significant sentences before or after generating new ones might lead to the omission of certain concepts, impacting the overall performance of the summaries and resulting in lower ROUGE scores. Contrary to common assumptions, longer sentences do not consistently equate to better summaries, nor do shorter sentences guarantee more informative summaries. To address these limitations, a potential approach to enhance the selection of the newly generated summary sentences would involve devising an optimization function to identify the best-performing sentences. One avenue for improving EXABSUM$_{\text{Abstractive}}$ could involve leveraging the optimal EXABSUM$_{\text{Extractive}}$ combination (Comb3) to achieve this objective.

Qualitative evaluation

Table 5 presents our qualitative evaluation, designed to assess user satisfaction with the produced summaries. When examining the varying percentages of assessed summaries within each category, we observe a moderate number of abstractive summaries that have received agreement compared to extractive summaries evaluated under the same criteria.

The information presented in the summaries generated using EXABSUM$_{\text{Abstractive}}$ was assessed positively in contrast to the extractive technique, and in terms of human perception, the abstractive summaries surpass the extractive ones in terms of quality. Additionally, it is noteworthy that the utilization of EXABSUM$_{\text{Abstractive}}$ leads to a reduction in the proportion of summaries receiving lower scores (strongly disagree and disagree). Table 6 illustrates an example of two summaries produced by EXABSUM$_{\text{Extractive}}$ and EXABSUM$_{\text{Abstractive}}$, respectively. As evident, certain sentences are shared by both summaries, while others have been truncated in the latter."

Table 5 Results of user satisfaction for various text summarization approaches

%	TS approach	Q1	Q2	Q3
1. Strongly disagree	EXABSUM$_{\text{Extractive}}$	8.76	17.25	17.25
	EXABSUM$_{\text{Abstractive}}$	1.44	0	1.21
2. Disagree	EXABSUM$_{\text{Extractive}}$	38.6	19.01	30.1
	EXABSUM$_{\text{Abstractive}}$	28.37	18.41	27.61
3. Neither agree nor disagree	EXABSUM$_{\text{Extractive}}$	21.39	17.44	22.83
	EXABSUM$_{\text{Abstractive}}$	19.2	17.44	22.83
4. Agree	EXABSUM$_{\text{Extractive}}$	19.2	48.06	42.1
	EXABSUM$_{\text{Abstractive}}$	50.46	34.83	5.32
5. Strongly agree	EXABSUM$_{\text{Extractive}}$	6.29	13.78	22.2
	EXABSUM$_{\text{Abstractive}}$	6.29	13.78	8.55

Table 6 Example summaries generated by EXABSUMExtractive and EXABSUMAbstractive for Document WSJ891019-0021 (DUC 2002 Corpus, Cluster d062j) with 50% ratio of original text

$EXABSUM_{Extractive}$
The White House is making sure nobody will accuse it of taking this crisis lightly
President Bush and his aides flew into a whirlwind of earthquake related activity yesterday morning
Mr Bush and his aides were accused of responding too slowly after the Exxon Valdez oil tanker split open in Alaskan waters and Hurricane Hugo struck the Carolina coast
Mr. Bush got his first earthquake briefing of the day at 6:30 a.m
Mr Bush said that he hoped there would be less carping about the emergency office performance this time adding that the agency took a hit for its reaction to Hurricane Hugo
$EXABSUM_{Abstractive}$
president bush visiting the california earthquake site this weekend. president bush and his aides flew into a whirlwind of earthquake related activity yesterday morning to get federal help flowing to victims, designed mostly to project image of white house in action, were trying to head off criticism, were accused of responding too slowly after the exxon valdez oil tanker split open in alaskan waters and hurricane hugo struck the carolina coast visited fema headquarters do not want a repeat those charges. the white house is making sure nobody will accuse it of taking crisis lightly

Comparison to baselines

In this subsection, we will compare the top results achieved by $\text{EXABSUM}_{\text{Extractive}}$ and $\text{EXABSUM}_{\text{Abstractive}}$ in generating single-document summaries with the performance of various state-of-the-art summarization techniques. Specifically, we will compare our summarization outcomes with:

I. The best-performing participants in the DUC 2001 and 2002 shared tasks.

II. The three most successful summarizers identified in a prior evaluation as documented in [63].

III. Other approaches, both recent and earlier, that utilized the DUC01 and DUC02 datasets, and evaluated their results using ROUGE-1 and ROUGE-2 metrics. The following subsections provide a brief overview of these approaches:

- Parveen and Strube [19] introduced an unsupervised graph-based technique for single-document summarization, which considers three essential summary features: significance, non-redundancy, and local coherence.
- Autosummarizer [64] is a web service that generates summaries by segmenting and ranking the most crucial sentences. Its single-document summarization method involves selecting the most pertinent sentences from the source document and has demonstrated superior performance compared to other summarizers in previous evaluations [65]. Unfortunately, details regarding the functioning of this system are not available.
- Classifier4J [66] is a text summarization and classification toolbox. It performs extractive single-document summarization based on word frequency and constructs the summary from the initial sentences that include any of the top-100 most frequent words in the document.
- UnifiedRank [67] is an approach that introduces an innovative unified method for single-document and multi-document summarization simultaneously. It utilizes a graph-based representation along with a unified ranking technique.

- DE [9] is a summarization technique based on sentence clustering. It optimizes the objective function using a discrete Differential Evolution method and similarity, thereby selecting representative sentences from each cluster.
- The Fuzzy Evolutionary Optimization Model (FEOM) [68] categorizes sentences based on document content and selects the most significant sentence from each cluster to represent the overall meaning of a document.
- NetSum [69] is a method that utilizes the RankNet learning algorithm to train a pair-based sentence ranker. It scores each phrase in a document to determine the most relevant sentences.
- Compendium [38]: a text summarization system used to generate two types of generic summaries—extractive and abstractive. It includes the variations COMPENDIUME and COMPENDIUME–A, where the former focuses on choosing and extracting the most relevant sentences using an extractive approach. The latter, COMPENDIUME–A, combines extractive and abstractive strategies by integrating an information compression and fusion stage to generate abstractive-oriented summaries.
- HP-UFPE Functional summarizing (HP-UFPE FS) [70]: A summary system that draws from seventeen extractive summarization methodologies that have garnered substantial attention in the literature, extensively explored in research papers, blogs, and news articles. In this evaluation, the HP-UFPE FS system is utilized, employing the optimal sentence scoring combination for news articles as detailed in [70].
- Get To The Point [71]: An abstractive summarization approach featuring coverage and utilizing a hybrid pointer-generator architecture. This technique addresses the challenge faced by conventional abstractive summarization systems on extensive documents, mitigating the generation of repeated and redundant words and phrases.
- Fast Abstractive Summarization [36]: Introduces a precise and efficient summarization model that initially selects important sentences and subsequently rewrites them in an abstractive manner—compressing and paraphrasing—to generate a concise final summary. The method employs a novel sentence-level policy gradient technique to connect the non-differentiable calculation between these two neural networks while maintaining linguistic fluency.

Tables 7, 8, and 9 provide a comparison between the top-performing configurations determined during our experiments and the summarizers mentioned earlier, focusing on the ROUGE-1, ROUGE-2, DUC 2001, DUC 2002, and CNN collections, respectively. Beginning with the DUC 2001 dataset, our systems (EXABSUM$_{\text{Extractive}}$ and EXABSUM$_{\text{Abstractive}}$) surpass the DE and FEOM systems in terms of both ROUGE-1 and ROUGE-2 scores (see Fig. 7).

Upon analyzing the feature weights derived from DE and FEOM, it becomes evident that both methods employ semantic features to ascertain the significance of sentences. This suggests that semantic techniques play a substantial role in the text summarization process. System T, which stands as the top-performing participant in the DUC 2001 competition, achieved superior ROUGE-2 results. However, it's worth noting that its performance is statistically similar to the outcomes produced by DE, FEOM, and Classifier4J (a supervised approach).

Table 7 F-measure comparison: our proposed techniques vs. baselines for single-document summarization on the DUC 2001 collection

Summary type	**Summary type**	**DUC 2001**	
		ROUGE-1	**ROUGE-2**
$\text{EXABSUM}_{\text{Extractive}}$	Extractive	**0.480**	**0.208**
$\text{EXABSUM}_{\text{Abstractive}}$	Abstractive	0.319	0.151
DE	Extractive	0.478	0.185
FEOM	Extractive	0.477	0.185
NetSum	Extractive	0.464	0.176
UnifiedRank	Extractive	0.453	0.176
System T (best DUC 2001 participant)	Extractive	0.445	0.202
Classifier4J	Extractive	0.444	0.198
Autosummarizer	Extractive	0.419	0.169
HP-UFPE FS	Extractive	0.359	0.117

The bold values emphasize the superior significance of our approach compared to others

Table 8 Comparison of F-measure results between our proposed techniques and the baseline methods on the DUC 2002 collection for the single-document summarization task

Systems	**Summary Type**	**DUC 2002**	
		ROUGE-1	**ROUGE-2**
$\text{EXABSUM}_{\text{Extractive}}$	Extractive	0.493	0.257
$\text{EXABSUM}_{\text{Abstractive}}$	Abstractive	0.341	0.104
Parveen and Strube [19]	Extractive	0.485	0.230
UnifiedRank	Extractive	0.484	0.214
System 28 (best DUC 2002 participant)	Extractive	0.480	0.228
Classifier4J	Extractive	0.470	0.221
DE	Extractive	0.466	0.123
FEOM	Extractive	0.465	0.124
COMPENDIUM E	Extractive	0.456	0.202
NetSum	Extractive	0.449	0.111
Autosummarizer	Extractive	0.437	0.191
COMPENDIUM E–A	Abstractive	0.395	–
Fast Abstractive Summarization	Abstractive	0.394	0.173
Get To The Point	Abstractive	0.372	0.157
HP-UFPE FS	Extractive	0.359	0.117

Table 9 F-measure results of our proposed approaches compared to baselines on the CNN dataset for single-document summarization task

Systems	**Summary Type**	**CNN**	
		ROUGE-1	**ROUGE-2**
$\text{EXABSUM}_{\text{Extractive}}$	Extractive	0.601	0.451
$\text{EXABSUM}_{\text{Abstractive}}$	Abstractive	0.501	0.332
HP-UFPE FS	Extractive	0.507	0.345
Autosummarizer	Extractive	0.488	0.327
Classifier4J	Extractive	0.466	0.321

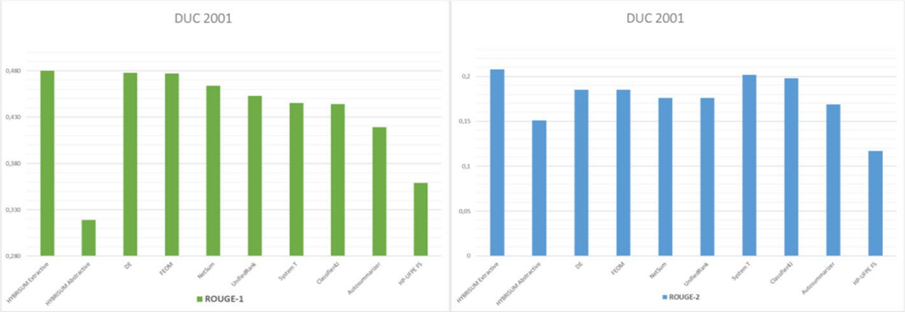

Fig. 7 Comparison of ROUGE-1 and ROUGE-2 Results: EXABSUM$_{\text{Extractive}}$ and EXABSUM$_{\text{Abstractive}}$ vs. various baseline systems on the DUC 2001 dataset

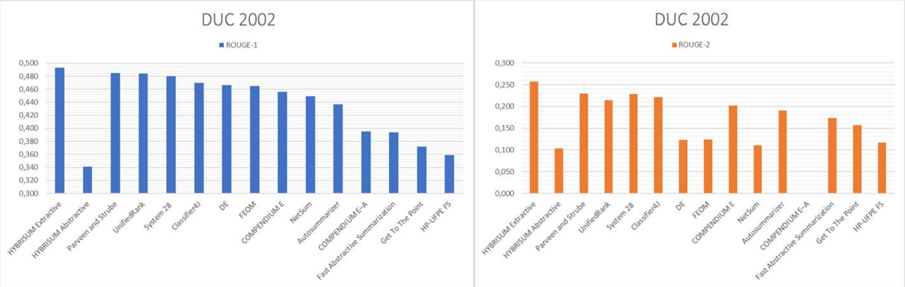

Fig. 8 ROUGE-1 and ROUGE-2 results of EXABSUMExtractive and EXABSUMAbstractive compared to various baseline systems on the DUC 2002 dataset

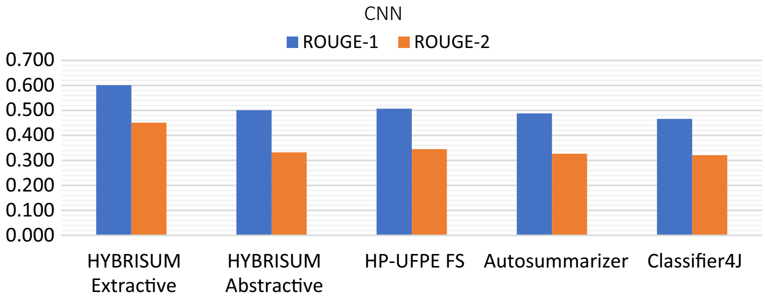

Fig. 9 ROUGE-1 and ROUGE-2 results of EXABSUM$_{\text{Extractive}}$ and EXABSUM$_{\text{Abstractive}}$ compared to various baseline systems on the large CNN dataset

On the DUC 2002 dataset, the top three performing systems are EXABSUM$_{\text{Extractive}}$, Parveen, and Strube [19] (See Fig. 8). It is worth noting that the Parveen and Strube [19] approach treats summarization as an optimization task, with an optimization step used to verify non-redundancy and local coherence in the resulting summaries. As expected, incorporating coherence and redundancy elimination approaches improves ATS performance. Despite the fact that the DUC2001 and DUC2002 contests have been running for a decade, the System T and System 28 still produce competitive results when compared to certain current summarizers. In contrast, while using deep learning methodologies, the ’get to the point’ and ’rapid abstractive summarization' systems (state-of-the-art abstractive methods) produced unimpressive results.

Regarding the CNN dataset, once more, $\text{EXABSUM}_{\text{Extractive}}$ emerges as the top performer in terms of ROUGE-1 and ROUGE-2 scores (see Fig. 9). Statistically, it surpasses all other systems, showcasing a remarkable 34.12% enhancement over the best-performing system.

Overall, our two automatic text summarization (ATS) techniques, namely $\text{EXABSUM}_{\text{Extractive}}$ and $\text{EXABSUM}_{\text{Abstractive}}$, demonstrate effectiveness in extractive and abstractive document summarization, respectively. They stand on par with other state-of-the-art text summarization tools, both extractive and abstractive. However, it's important to note that $\text{EXABSUM}_{\text{Abstractive}}$ falls short compared to $\text{EXABSUM}_{\text{Extractive}}$. The employment of $\text{EXABSUM}_{\text{Extractive}}$ results in higher ROUGE-1 scores when compared to other techniques. The performance of $\text{EXABSUM}_{\text{Abstractive}}$ lags behind $\text{EXABSUM}_{\text{Extractive}}$ across most metrics, as it solely relies on the input content and lacks the enhancement provided by the most relevant extractive sentences produced by $\text{EXABSUM}_{\text{Extractive}}$.

As mentioned earlier, the ROUGE evaluation relies on exact matches of text fragments when comparing system-generated summaries to human-produced ones (extracts). Consequently, if abstractive information is integrated with the extractive output summary in a hybrid model, it's possible that the F-measure results could significantly improve compared to the initial extract. This insight suggests that further research could be conducted into these types of summaries, aiming to enhance their quality beyond the mere selection of sentences.

In our specific approach to creating and testing $\text{EXABSUM}_{\text{Abstractive}}$, we employed two methods: (1) We initiated the process by crafting abstractive summaries based on sentences identified as relevant during the sentence relevance stage. These summaries underwent compression or merging of information, followed by reranking. This approach led to resulting summaries that were shorter than the extracts produced by $\text{EXABSUM}_{\text{Extractive}}$. Since no additional information was introduced, the recall value was consistently lower than that of $\text{EXABSUM}_{\text{Extractive}}$. (2) For the second technique, we utilized $\text{EXABSUM}_{\text{Abstractive}}$ to generate new sentences from the source document, starting with the first word of each sentence. Ultimately, we opted for this latter technique, as it appeared more suitable for producing more accurate summaries. However, it's important to note that further research efforts are required to enhance $\text{EXABSUM}_{\text{Abstractive}}$, including exploring techniques such as rephrasing and embedding.

Based on the ROUGE evaluation results presented in Tables 7, 8, and 9, it's evident that the proposed summarization approach, EXABSUM, demonstrates strong performance across its two variants: $\text{EXABSUM}_{\text{Extractive}}$ and $\text{EXABSUM}_{\text{Abstractive}}$. These findings underscore the significance of both types of summaries, as they collectively contribute to the creation of informative summaries, ultimately enhancing the performance of the text summarization task. Additionally, our proposed approach for automatic text summarization effectively aids in selecting sentences that are not only informative but also grammatically correct and semantically relevant to the text.

Summary and conclusions

In this paper, we introduced EXABSUM, a novel approach to automatic text summarization (ATS) specifically designed for single-document summarization. EXABSUM offers the generation of two types of summaries: extractive and abstractive, represented by the corresponding $\text{EXABSUM}_{\text{Extractive}}$ and $\text{EXABSUM}_{\text{Abstractive}}$ variants. The extractive technique integrates statistical and semantic scoring methods, including the innovative 'TR-ISF' approach, to effectively select and extract non-redundant and relevant sentences. On the other hand, the abstractive approach incorporates information compression, fusion, and re-ranking strategies based on keyphrases, producing abstractive summaries from the source document rather than relying solely on the extractive summary.

The evaluation results showcased the effectiveness of our two ATS techniques. They demonstrated their ability to capture crucial information across various domains through benchmark datasets. With the proposed architecture, EXABSUM not only retains essential information but also generates coherent and distinct abstractive summaries alongside the extractive ones. Furthermore, the modular structure of EXABSUM facilitates easy integration of new phases, allowing for potential improvements and expanded functionalities of the ATS method.

This promising result suggests the avenue for future research in abstractive and hybrid ATS approaches to enhance their performance beyond mere sentence selection. Our work highlights the potential of this novel direction for the NLP community. In the future work, we plan to assess EXABSUM's performance using hybrid techniques and explore alternative methods to enhance abstractive summaries, including rephrasing techniques and integration of deep learning methodologies.

Abbreviations

IR	Information retrieval
NLP	Natural language processing
ATS	Automatic text summarization
ROUGE	Recall-oriented understudy for gisting evaluation
HFSM	Hybrid feature selection model

Acknowledgements
Not applicable.

Author contributions

Author ZAM is the first author who: conceived and designed the analysis; collected the data; contributed data or analysis tools; performed the analysis; wrote the paper. Professors BF and BO also contribute to the paper in conceiving and designing the analysis, helping the whole methodology findings. All authors reviewed the manuscript.

Funding
Not applicable.

Availability of data and materials

The data was collected from the case company and is not available to the general public. The authors' data are, however, available upon reasonable request and with the permission of the case study company. The used datasets are available in:DUC dataset: https://duc.nist.gov/data.html; CNN dataset: https://sites.google.com/view/summarizationcorpus/p%C3%A1gina-inicial. Requests for software application or custom code should be made to the corresponding authors upon reasonable request.

Declarations

Ethics approval and consent to participate

This article does not contain any studies with human participants or animals performed by any authors.

Consent for publication

Not applicable.

Competing interests

The authors declare that they have no competing interests. All authors certify that they have no affiliations with or involvement in any organization or entity with any financial interest or non-financial interest in the subject matter or materials discussed in this manuscript.

Received: 18 August 2022 Accepted: 3 October 2023
Published online: 24 October 2023

References

1. Hovy E, Marcu D. Automated text summarization. The Oxford handbook of computational linguistics. 2005, pp. 583–598.
2. Mani I, Maybury MT. Advances in automatic text summarization. Cambridge: The MIT Press; 1999.
3. Huang L, He Y, Wei F, Li W. Modeling document summarization as multi-objective optimization. In: Proceedings of the third international symposium on intelligent information technology and security informatics. 2010, pp 382–386.
4. Gupta S, Gupta SK. Abstractive summarization: an overview of the state of the art. Expert Syst Appl. 2019;121:49–65.
5. Nenkova A, & McKeown K. A survey of text summarization techniques. In Mining text data. Springer; 2012, pp. 43–76.
6. Luhn HP. The automatic creation of literature abstracts. IBM J Res Dev. 1958;2(2):159–65.
7. Barrios F, López F, Argerich L et al. Variations of the similarity function of textrank for automated summarization. The Argentine Symposium on Artificial Intelligence (ASAI) 2015-44 JAIIO; 44 JAIIO-ASAI 2015-ISSN: 2451–7585, 2016. pp 65–72.
8. Dagan I, Marcus S, Markovitch S. Contextual word similarity and estimation from sparse data. In: Proceedings of the 31st annual meeting on Association for Computational Linguistics, pp. 164–171. Association for Computational Linguistics (1993).
9. Aliguliyev RM. A new sentence similarity measure and sentence based extractive technique for automatic text summarization. Expert Syst Appl. 2009;36(4):7764–72.
10. Alcón O, Lloret E. SEMPCA-Summarizer: exploiting semantic principal component analysis for automatic summary generation. Comput Informs. 2018;37:1126–48.
11. Erkan G, Radev DR. Lexrank: graph-based lexical centrality as salience in text summarization. J Artif Intell Res. 2004;22:457–79.
12. Radev D, Allison T, Blair-Goldensohn S, Blitzer J, Celebi A, Drabek E, Lam W, Liu D, Otterbacher J, Qi H, Saggion H, Teufel S, Topper M, Winkel A, Zhang Z. MEAD—a platform for multidocument multilingual text summarization, Proceedings of the 4th International Conference on Language Resources and Evaluation, 2004, pp. 699–702.
13. Dunlavy DM, O'Leary DP, Conroy JM, et al. QCS: a system for querying, clustering and summarizing documents. Info Process Manag. 2007;43(6):1588–605.
14. Saggion H, Poibeau T. Automatic text summarization: past, present and future. In: Multi-source, multilingual information extraction and summarization. Springer, Berlin, Heidelberg; 2013. p. 3–21.
15. Liu X, Webster JJ, Kit C. An extractive text summarizer based on significant words. In: Proceedings of the 22nd international conference on computer processing of oriental languages, language technology for the knowledge-based economy, Springer; 2009. pp 168–178.
16. Tonelli S, Pianta E. Matching documents and summaries using key concepts. In: Proceedings of the French text mining evaluation workshop. 2011.
17. Ko Y, Seo J. An effective sentence-extraction technique using contextual information and statistical approaches for text summarization. Pattern Recognit Lett. 2008;29:1366–71. https://doi.org/10.1016/j.patrec.2008.02.008.
18. Baralis E, Cagliero L, Mahoto N, Fiori A. GRAPHSUM: discovering correlations among multiple terms for graph-based summarization. Inf Sci. 2013;249:96–109. https://doi.org/10.1016/j.ins.2013.06.046.
19. Parveen D, Strube M. Integrating importance, non-redundancy and coherence in graph-based extractive summarization. In: Proceedings of the 24th international conference on artificial intelligence. AAAIPress; 2015. pp 1298–1304.
20. Durrett G, Berg-Kirkpatrick T, Klein D. Learning-based single-document summarization with compression and anaphoricity constraints. In Proceedings of the 54th annual meeting of the association for computational linguistics, Volume 1: Long Papers; 2016. pp. 1998–2008.
21. Alguliev RM, Aliguliyev RM, Hajirahimova MS, Mehdiyev CA. MCMR: maximum coverage and minimum redundant text summarization model. Expert Syst Appl. 2011;38:14514–22. https://doi.org/10.1016/j.eswa.2011.05.033.
22. Lin H, Bilmes J. Multi-document summarization via budgeted maximization of submodular functions. In: Human language technologies: the 2010 annual conference of the North American chapter of the association for computational linguistics, Association for Computational Linguistics, 2010. pp 912–920.
23. Yao JG, Wan X, Xiao J. Phrase-based compressive cross-language summarization. In: Proceedings ofthe 2015 conference on empirical methods in natural language processing; 2015. pp 118–127.
24. Plaza L. Uso de Grafos Semánticos en la Generación Automática de Resúmenes y Estudio de su Aplicación en Distintos Dominios: Biomedicina, Periodismo y Turismo, PhD thesis, 2011.
25. Belz A. Automatic generation of weather forecast texts using comprehensive probabilistic generation-space models. Nat Lang Eng. 2008;14(4):431–55.
26. Mohammad S, Dorr B, Egan M, Hassan A, Muthukrishan P, Qazvinian V, Radev D, Zajic D. Using citations to generate surveys of scientific paradigms, Proceedings of the North American Chapter of the Association of Computational Linguistics, 2009, pp. 584–592.
27. Erera S, Shmueli-Scheuer M, Feigenblat G, Nakash OP, Boni O, Roitman H, et al. A summarization system for scientific documents. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the

9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP): System Demonstrations, 2019, pp. 211–216.

28. Barzilay R, McKeown KR. Sentence fusion for multidocument news summarization. Comput Linguist. 2005;31(3):297–328.
29. Filippova K, Strube M. Sentence fusion via dependency graph compression. In Proceedings of the 2008 conference on empirical methods in natural language processing, Honolulu, Hawaii, October; 2008. Association for Computational Linguistics. pp 177–185.
30. Filippova K. Multi-sentence compression: finding shortest paths in word graphs. In: Proceedings of the 23rd international conference on computational linguistics. Association for Computational Linguistics, 2010. p. 322–330.
31. Mahajani A, Pandya V, Maria I, Sharma D. A comprehensive survey on extractive and abstractive techniques for text summarization. Paper presented at the Ambient Communications and Computer Systems, Singapore. 2019.
32. Boudin F, Morin E. Keyphrase extraction for n-best reranking in multi-sentence compression. In Proceedings of the 2013 conference of the North American chapter of the association for computational linguistics: human language technologies, Atlanta, Georgia, June. Association for Computational Linguistics. 2013. pp 298–305.
33. Banerjee S, Mitra P, Sugiyama K. Multi-document abstractive summarization using ilp based multi-sentence compression. In Proceedings of the 24th International Conference on Artificial Intelligence, IJCAI'15. 2015. p. 1208–1214. AAAI Press.
34. Nayeem MT, Fuad TA, Chali Y. Abstractive unsupervised multi-document summarization using paraphrastic sentence fusion. In: Proceedings of the 27th International Conference on Computational Linguistics. 2018. p. 1191–1204.
35. Shang G, Ding W, Zhang Z, Tixier AJP, Meladianos P, Vazirgiannis M, Lorré JP. Unsupervised abstractive meeting summarization with multi-sentence compression and budgeted submodular maximization. In ACL (1). 2018.
36. Chen YC, Bansal M. Fast abstractive summarization with reinforce-selected sentence rewriting. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, Vol. 1: Long Papers, 2018. pp. 675–686.
37. Di Fabbrizio G, Stent A, Gaizauskas R. A hybrid approach to multi-document summarization of opinions in reviews. In: Proceedings of the 8th International Natural Language Generation Conference (INLG). 2014. p. 54–63.
38. Lloret E, Romá-Ferri MT, Palomar M. COMPENDIUM: a text summarization system for generating abstracts of research papers. Data Knowl Eng. 2013;88:164–75.
39. Bhat IK, Mohd M, Hashmy R. SumItUp: a hybrid single-document text summarizer. In Pant M, Ray K, Sharma TK, Rawat S, Bandyopadhyay A (eds.) Soft computing: theories and applications: proceedings of SoCTA 2016, Vol. 1. Singapore: Springer Singapore; 2018. pp. 619–634.
40. De Marneffe MC, MacCartney B, Manning CD, et al. Generating typed dependency parses from phrase structure parses. In: Lrec, 2006;6:449–454.
41. Glickman O. Applied textual entailment challenge. Ph.D. thesis, Bar Ilan University. 2005.
42. Tatar D, Mihis AD, Lupsa D. Text entailment for logical segmentation and summarization. Proceedings of the 13th International Conference on Applications of Natural Language to Information Systems, Lecture Notes in Computer Science, vol. 5039, Springer, 2008, pp. 233–244.
43. Parikh A, Täckström O, Das D, Uszkoreit J. A decomposable attention model for natural language inference. In Proceedings of the 2016 conference on empirical methods in natural language processing; 2016. pp. 2249–2255.
44. Pasunuru R, Bansal M. Multi-reward reinforced summarization with saliency and entailment. In Proceedings of the 2018 Conference of the North American chapter of the association for computational linguistics: human language technologies, Vol. 2 (Short Papers); 2018. pp. 646–653.
45. Lloret E, Palomar M. A gradual combination of features for building automatic summarization systems. In Proceedings of the 12th international conference on text. Speech and dialogue. Berlin, Heidelberg: Springer-Verlag; 2009. pp. 16–23.
46. Ferrández ´O. Textual entailment recognition and its applicability in NLP tasks. PhD thesis, University of Alicante; 2009.
47. Edmundson HP. New methods in automatic extracting. J ACM. 1969;16(2):264–85.
48. Ferreira R, de Souza Cabral L, Lins RD, Pereira e Silva G, Freitas F, Cavalcanti GD, et al. Assessing sentence scoring techniques for extractive text summarization. Expert Syst Appl. 2013;40(14):5755–64.
49. Ouyang Y, Li W, Lu Q, Zhang R. A study on position information in document summarization. In Proceedings of the 23rd international conference on computational linguistics: Posters. COLING '10. Stroudsburg, PA, USA: Association for Computational Linguistics; 2010. pp. 919–927.
50. Abuobieda A, Salim N, Albaham AT, Osman AH, Kumar YJ. Text summarization features selection method using pseudo genetic-based model. In Proceedings of the international conference on information retrieval & knowledge management. 2012. pp. 193–197.
51. Fattah MA, Ren F. GA, MR, FFNN, PNN and GMM based models for automatic text summarization. Comput Speech Lang. 2009;23(1):126–44.
52. Li Y, Luo C, Chung SM. Text clustering with feature selection by using statistical data knowledge and data engineering. IEEE Trans Knowl Data Eng. 2008;20(5):641–51.
53. Benghabrit A, Ouhbi B, Frikh B, Behja H. Text clustering using statistical and semantic data. In Proceedings of the 2013 World Congress on Computer and Information Technologies, 2013, 1–6.
54. Oliveira H, Ferreira R, Lima R, Lins RD, Freitas F, Riss M, Simske SJ. Assessing shallow sentence scoring techniques and combinations for single and multi-document summarization. Expert Syst Appl. 2016;65:68–86.
55. Merrouni ZA, Frikh B, Ouhbi B. Automatic keyphrase extraction: a survey and trends. J Intell Inf Syst. 2019; p. 1–34. Springer.
56. Wan X, Xiao J. Collabrank: towards a collaborative approach to single-document keyphrase extraction. In Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008), pages 969–976, Manchester, UK, August. Coling 2008 Organizing Committee.
57. Mihalcea R, Tarau P. Textrank: bringing order into texts. In Dekang Lin and Dekai Wu, editors, Proceedings of EMNLP 2004, pages 404–411, Barcelona, Spain, July. Association for Computational Linguistics. 2004.

58. Lins RD, Oliveira H, Cabral L, Batista J, Tenorio B, Ferreira R, et al. The cnn-corpus: a large textual corpus for single-document extractive summarization. In Proceedings of the ACM Symposium on Document Engineering 2019. 2019, pp. 1–10.
59. Lins RD, Ferreira R, Simske SJ. DocEng'19 Competition on Extractive Text Summarization. In Proceedings of the 2019 ACM Symposium on Document Engineering (DocEng '19). ACM, New York, NY, USA, 2019. pp 216–217. https://doi.org/10.1145/3342558.3351874
60. Lin CY. Rouge: A package for automatic evaluation of summaries. In Text summarization branches out, 2004. pp. 74–81.
61. Lin C-Y, Hovy E. Automatic evaluation of summaries using n-gram co-occurrence statistics. In: Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics. 2003. p. 150–157.
62. Ferrández O, Micol D, Muñoz R, Palomar M. A perspective-based approach for solving textual entailment recognition. Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing, 2007, pp. 66–71.
63. Cao Z, Li W, Li S, Wei F. Improving multi-document summarization via text classification. In Thirty-First AAAI Conference on Artificial Intelligence. 2017.
64. Autosummarizer. 2015. Retrieved from http://autosummarizer.com/.
65. Batista J, Ferreira R, Tomaz H, Ferreira R, Dueire Lins R, Simske S. A quantitative and qualitative assessment of automatic text summarization systems. In Proceedings of the 2015 ACM Symposium on Document Engineering, 2015. pp. 65–68.
66. Classifier4J. 2005. Retrieved from http://classifier4j.sourceforge.net/.
67. Wan X. Towards a unified approach to simultaneous single-document and multi-document summarizations. In Proceedings of the 23rd international conference on computational linguistics (Coling 2010), pp. 1137–1145.
68. Song W, Choi LC, Park SC, Ding XF. Fuzzy evolutionary optimization modeling and its applications to unsupervised categorization and extractive summarization. Expert Syst Appl. 2011;38(8):9112–21.
69. Svore K, Vanderwende L, Burges C. Enhancing single-document summarization by combining RankNet and third-party sources. In Proceedings of the 2007 joint conference on empirical methods in natural language processing and computational natural language learning (EMNLP-CoNLL), 2007. pp. 448–457.
70. Ferreira R, de Freitas FLG, de Souza Cabral L, Lins RD, Lima R, de França Pereira e Silva G, et al. A context-based text summarization system. In Proceedings of the 11th international workshop on document analysis systems (das), 2014. pp. 66–70.
71. See A, Liu PJ, Manning CD. Get to the point: summarization with pointer-generator networks. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, Vol. 1: Long Papers; 2017. pp. 1073–1083.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.