

Received October 27, 2021, accepted November 18, 2021, date of publication November 22, 2021, date of current version November 30, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3129786

A Survey of Automatic Text Summarization: Progress, Process and Challenges

M. F. MRIDHA¹, (Senior Member, IEEE), AKLIMA AKTER LIMA¹,
KAMRUDDIN NUR², (Senior Member, IEEE), SUJOY CHANDRA DAS¹,
MAHMUD HASAN³, AND MUHAMMAD MOHSIN KABIR¹

¹Department of Computer Science and Engineering, Bangladesh University of Business and Technology, Dhaka 1216, Bangladesh

²Department of Computer Science, American International University-Bangladesh, Dhaka 1229, Bangladesh

³Department of Computer Science, Western University, London, ON N6A 3K7, Canada

Corresponding author: M. F. Mridha (firoz@bubt.edu.bd)

ABSTRACT With the evolution of the Internet and multimedia technology, the amount of text data has increased exponentially. This text volume is a precious source of information and knowledge that needs to be efficiently summarized. Text summarization is the method to reduce the source text into a compact variant, preserving its knowledge and the actual meaning. Here we thoroughly investigate the automatic text summarization (ATS) and summarize the widely recognized ATS architectures. This paper outlines extractive and abstractive text summarization technologies and provides a deep taxonomy of the ATS domain. The taxonomy presents the classical ATS algorithms to modern deep learning ATS architectures. Every modern text summarization approach's workflow and significance are reviewed with the limitations with potential recovery methods, including the feature extraction approaches, datasets, performance measurement techniques, and challenges of the ATS domain, etc. In addition, this paper concisely presents the past, present, and future research directions in the ATS domain.

INDEX TERMS Automatic text summarization, feature extraction, summarization methods, performance measurement matrices, challenges.

I. INTRODUCTION

The amount of textual material on the web and other libraries is growing tremendously daily. Information utilization has become an expensive and time-consuming activity since data expands in a large quantity at a time and includes irrelevant content or noise. Text summarization is a method used to summarize the data. A manual text summarization process is undoubtedly an effective way to preserve the meaning of the text; however, this is a time-consuming activity. Another approach is to utilize the *automatic text summarization (ATS)*. In ATS, different practical algorithms can be programmed into computers to produce summaries of information. Thus, text summarization creates a brief and accurate overview of a lengthy text document by concentrating on the essential parts that provide valuable details by maintaining the overall context. In natural language processing (NLP), automatic text summarization is a method of evaluating, comprehending, and extracting information from human language. Nowadays,

students to researchers, business leaders to business analysts, people from every domain work with numerous documents. Sometimes, people get confused about finding the relevant part within a document or documents where ATS can be very helpful and useful. The fundamental goal of ATS is to create a compact and persuasive summary and maintain critical information from the document. ATS also aims to generate a review that condenses the significant ideas from the input content into a small amount of space. Furthermore, the ATS systems assist users in obtaining the essential points of the original content without reading the complete document. Users will profit from the automatically generated summaries, saving them a great deal of time and work.

The objectives of this study focus on providing a thorough review of various ATS research projects. To acquire deep knowledge, researchers require a sense of what has already been done and further possibilities in this broad topic. Therefore, this study aims to assist academics and professionals in developing an idea of the evolution of ATS, research progress, and future research directions in

The associate editor coordinating the review of this manuscript and approving it for publication was Lefei Zhang¹.

this topic. In addition, the apparent obstacles or limitations in future research in this field are also discussed in this paper.

Text summarization was invented by H.P. Luhn [1] in the 1950s, which was used in the first commercial computer IBM 701. Using the bag-of-words approach, he counted the frequencies of the most frequent words based on their occurrence. Then, the most frequent words were selected and assigned a number to each sentence depending on a regular event. Gradually, linguistics was being considered and began to use various word types and formation using natural language processing (NLP). The extraction, categorization, and classification of texts are the main targets. After that, the evolution of NLP between 1990 and 2000 introduced the conversion of sentences into vectors and words into their base forms. The introduction of advanced NLP techniques such as neural word embedding [2], Bag of Words (BoW) [3], and word2vec [4], and modern deep learning approaches such as recurrent neural networks (RNN) [5] and long short-term memory (LSTM) [6] have observed significant progress in the ATS domain. The evolution of ATS from the 1950s to the present is reviewed in this study.

Text summarization processes from the 1970s to the early 2000s are considered traditional methods. Traditional text summarization processes require a better knowledge of the document to find the essential keywords. ATS has become an appealing domain for its influential assistance in the study and expansion of automation, too [7]. The improvement behind this new ATS is achieved by following a standard structure. Text summarization becomes more accurate and fluent by getting trimmed and interpreted with the proper design of processes.

As we have investigated ATS in-depth in this comprehensive survey, the study required a collection of scholarly research between 1998 and 2021. We followed a systematic literature review (SLR) approach to complete the review. Kitchenham proposed this systematic literature review (SLR) approach [8], [9] which consists of three phases: planning, conducting, and reporting the review. The SLR approaches tried to answer all possible questions that could arise while progressing in this research field. The goal of this research was to examine the findings of several essential research disciplines. The necessary materials for this research are assembled using the (Preferred Reporting Items for Systematic Reviews and Meta-Analyses) workflow diagram. The PRISMA workflow for this survey is shown in Figure 1.

The overall contributions of this paper are given as follows:

- This article performs a systematic review of the automatic text summarization, including the fundamental theories and evolutions.
- The survey includes the investigation of the existing dataset, feature extraction, text summarization approaches, text summarization algorithms, performance measurement, evaluation matrices, and challenges.

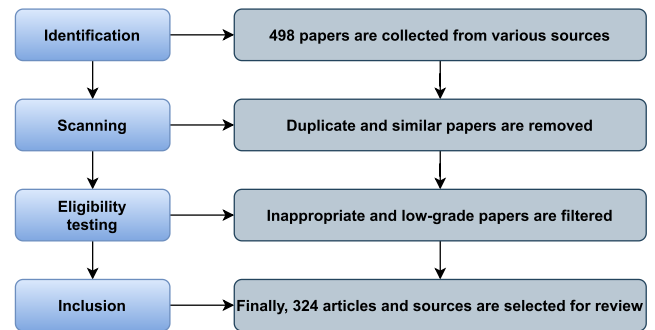


FIGURE 1. The image explains the PRISMA flow diagram for this literature survey. Number of papers collected during the systematic literature review are mentioned in the PRISMA statement.

- The article compiles ATS architectures based on current methods, datasets, feature extraction, and summarization approaches. Moreover, this study explains the constraints and limitations of such methods.
- Subsequently, the study ends by distinguishing the current difficulties and challenges of ATS architectures, along with future research directions.

The remainder of this paper is organized as follows: The literature review of existing ATS surveys shown in Section II, the motivations and applications of ATS are described in Section III, the basic structure is provided in Section IV, the most commonly used datasets in ATS are described in Section V, the widely used pre-processing techniques are addressed in Section VI, the strategies for extracting features are described in Section VII, main ATS approaches are discussed in VIII and algorithms are described in Section IX. The ATS approaches are reviewed in Section X and the ATS measuring performance methods are discussed in Section XI. The ATS challenges with potential research objectives are addressed in Section XII. Finally, Section XIII concludes the paper.

II. LITERATURE REVIEW OF EXISTING ATS SURVEY

We have investigated the existing surveys of the ATS domain, and a few of them are presented to prove the significance of this paper. Most surveys covered the former methods and research on ATS. However, recent trends, applicability, effects, limitations, and challenges of ATS techniques were not present. Table 1 summarizes and compares the existing survey on ATS.

Mishra *et al.* [12] reviewed (2000-2013) years of studies and found some methods such as hybrid statistical and ML approaches. The researchers did not include cognitive aspects or evaluations of the impact of ATS. Allahyari *et al.* [15] investigated different processes such as topic representation, frequency-driven, graph-based, and machine learning methods for ATS. This research only includes the frequently used strategies. El-Kassas *et al.* [17] described graph-based, fuzzy logic-based, concept-oriented, ML approaches, etc., with their advantages or disadvantages. This research did not include abstractive or hybrid techniques. Saranyamol and Sindhu [11] offered a thorough survey

TABLE 1. Existing ATS methods, their scope and limitations.

Reference	Year	Main Purpose	Limitations
[10]	2009	Critical ways to summarize the texts and provides a taxonomy of the methods of summarization.	Extractive, abstract, NLP with Machine learning and Deep learning are missed.
[11]	2014	A hybrid approach can do both the extractive and abstractive methods efficiently.	Avoided complex processing like NLP approaches.
[12]	2014	Reviewed works between (2000-2013) year and proposed a hybrid statistical method.	This paper doesn't include cognitive aspects, including visualization techniques and evaluations of the impact.
[13]	2016	Describes two definite summarization techniques, which are abstractive, extractive.	Introduces techniques and methods only.
[14]	2017	A study based on automated keyword extraction and text summarization.	They do not briefly review every approach they included; they missed some feature extraction model.
[15]	2017	Topic Representation, frequency-driven, graph-based, and the effectiveness and Limitations.	The recent approaches are not surveyed.
[16]	2017	Processes of extractive methods and multilingual text summarization are discussed.	A precise classification and idea about feature scores and extraction is missing.
[17]	2020	Methods, processes, primary structure, strategies, datasets, measurements of ATS.	A detailed classification and description of feature extraction are missing.
[18]	2020	To handle multi- documents for summarization based on recent research work and comparison.	Does not represent any brief discussion of any topic.

for analysts by introducing various aspects of ATS such as structure, strategies, datasets, evaluation metrics, etc. Gambhir and Gupta [16] attempted to analyze a hybrid approach including two text summarization methods. This study missed many contemporary techniques for review. The research of Gholamrezazadeh *et al.* [10] represents a comprehensive and comparative study of extractive methods in ATS of the last decade. Several multilingual approaches have also been discussed. Andhale and Bewoor [13] provided a taxonomy of text summarization methods and a variety of techniques. Although the author has covered some time-consuming processes of ATS, recent, more efficient methods such as machine learning were missed. Abualigah *et al.* [18] conducted research on how to handle multiple documents and massive web data for text summarization. Lastly, the paper contains a comparative table with recent studies without details. Bharti and Babu [14] presented a survey of research papers based on automated keyword extraction methods and techniques. It covers ideas about multiple databases that are used for document summarization.

III. MOTIVATION AND APPLICATION OF ATS

This study aims to provide an overview of current research in NLPs and, precisely, ATS to accelerate knowledge about it. In addition, it allows the creation of new tools, methods, datasets, and resources that meet the needs of the research and industrial sectors. The advancement of NLPs made automatic text summarization usable for a regular text document summary and sentiment analysis. Moreover, ATS promotes a versatile approach to research various fields such as machine learning, natural language, cognitive science, and psychology. With multiple sources of information, ATS discusses cutting-edge work and future directions in this exciting area. These collective findings are the motivation behind this research. An essential part of research on ATS is the application, which is presented in the following section.

Recently ATS has extensively employed applications based on information retrieval, information extraction, question answering, text mining, and analytics. The ATS also improves

the search engine's capabilities with various applications, including news summary, email summarizations, domain-specific summarization. Now, the applications of the ATS domain are presented below:

- 1 Books or Novel Summarization:** ATS is used mainly to summarize long documents such as books, literature, or novels, as short documents are unsuitable for summarization. It is not easy to find context from short texts, whether long documents are a better summary material [19].
- 2 Social Posts or Tweet Summarization:** Every day, millions of messages, posts are generated on social networking sites such as Facebook, Twitter, etc. Useful important text summarization can be achieved using ATS [20]. This valuable source of information using the ATS [20].
- 3 Sentiment Analysis (SA):** The analysis of people's views, feelings, and judgments regarding events and situations is known as sentiment analysis. SA classifies emotions and mostly opinions from product reviews as "Positive" or "Negative" using fuzzy logic. ATS is quite helpful for market analysts in summarizing the feelings or thoughts of hundreds of people [21].
- 4 News Summarization:** The ATS helps summarize news from many websites, such as CNN and other prominent news portals. ATS extracts the primary emphasis point of the story in a newspaper, which is sometimes used as the story's headline [22].
- 5 Email Summarization:** Email communications are unstructured and not usually syntactically well-formed domains for summarization. ATS usually extracts noun phrases and generates a summary of email messages using linguistic methods, and machine learning algorithms [23].
- 6 Legal Documents Summarization:** ATS discovers relevant prior instances based on legal questions and rhetorical functions to summarize a legal judgment document. A hybrid approach employs various methods, including keywords, critical phrase matching, and case-based analysis [24].

- 7 **Biomedical Documents Summarization:** ATS combines genetic clustering and connectivity information with a graph-based summarization process. Genetic clustering identifies the various themes of a biological document, whereas connectivity data demonstrate the relative significance of the study [25].
- 8 **Summarization of Scientific Papers:** Scientific documents are well-structured texts with numerous researchers' viewpoints on a similar topic. In addition, the critical points of a scientific document are primarily found in tables and figures rather than generic texts. A multi-document ATS framework combines two methods to produce a technical survey of scientific documents. First, follow and gather the citations and finally, use summarization techniques to determine the content of the original and related cited articles [26].

IV. STRUCTURE OF ATS

The basic architecture of ATS contains several sections, including an input layer that consists of two types of documents, and there are single-documents and multi-documents, respectively. A single-document summarization (SDS) system that selects the important sentences from the source document while considering the maximum limit of the summary [27]. Whereas, in a Multi-document summarization (MDS), multiple documents are selected as input to generate the summary [28]. Generally, an automatic text summarization goes through a set of stages including pre-processing, feature extraction, application of summary generation algorithm or methods to summarize the source document. In a single document summarization system, the monolithic structure of the document is utilized, whereas, in a multi-document summarization system, the document's structure is less relied on. The fundamental issue with MDS arose from the aggregation of many resources from which the data is taken, which may contain more redundant information than is typically present in a single document. Furthermore, putting the collected data into a coherent text from multiple documents to make a cohesive summary [29]. Summarization on news [30], scientific publications [31], emails [32], [33], product reviews [34], lecture feedback [35], [36], Wikipedia article generation [37], medical documents [38], and software project activities [39] are just a few examples of real-world applications for the multi-document summarization task [28]. Summarization can be done in two ways: abstractive or extractive. The abstractive approach is more achievable than the extractive approach for achieving concise summarization of contrary views and preferences [40]. The details of both abstractive and extractive approaches are discussed in section VIII. An illustration of the basic structure and steps of an ATS is presented in Figure 2 and discussed as follows:

A. PRE-PROCESSING

In the pre-processing phase, the linguistic techniques are used to pre-process input text documents using crucial techniques such as sentence segmentation, punctuation marks

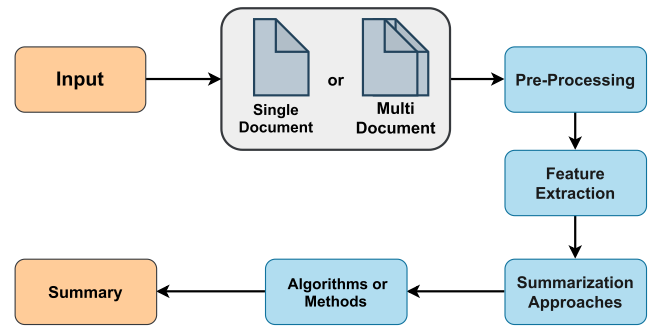


FIGURE 2. A basic structure of an ATS. The figure illustrates seven steps where the input (single or multi-document) are pre-processed and follows feature extraction. The next section follows summarization approaches and methods which concludes to a summary of the document.

removal, filtering stop-words, stemming (reducing common root words), etc. Widely used pre-processing techniques are discussed in Section VI.

B. FEATURE EXTRACTION

The extraction of sentences is vital for the entire summarization process by selecting different features within the source document. Selected features are applied to each sentence, and the highly scored sentences are chosen for the summary in feature extraction phase. Feature extraction techniques of ATS are elaborated in details in Section VII.

C. SUMMARIZATION APPROACHES

The first and most crucial step in the summarization strategy is to determine efficient methodologies. Some methods involve selecting the essential words and lines from the texts, while others involve paraphrasing one sentence and condensing the original content. Detailed text summarization approaches are discussed in Section VIII.

D. ALGORITHMS

Algorithms or methods are a more definite way of defining text summarization. Different algorithms and methods under various approaches are applied to obtain a better version of the summarized text. Text summarization algorithms are discussed in details in Section IX.

In the following section, we explore the important use of different widely used data sets in ATS application and research.

V. DATASETS USED IN AUTOMATIC TEXT SUMMARIZATION

This survey presents an overview of the essential resources used to analyze and assess the ATS fields. This section includes the datasets that are most often used for ATS evaluation. These text summarization datasets are mostly a collection of text pieces, articles, parts of newspapers, etc. These datasets are used in their respective sectors, such as newspaper datasets used only for newspaper summary purposes, as they have the same pattern. In addition, these datasets can be classified based on sources. Text

TABLE 2. The table presented the popular datasets used in the text summarization domain.

Dataset Name	Number of Documents / Sentences	Language	Used in	URL
New Taiwan Weekly	2738 sentences	English	[41]–[44]	http://www.newtaiwan.com.tw
Annotated English Gigaword	9,876,086 documents	English	[16], [45]–[50]	https://catalog.ldc.upenn.edu/LDC2011T07
DUC 2001	600 sentences	English	[51]–[56]	https://www-nlpir.nist.gov/projects/duc/data.html
DUC 2002	600 sentences	English	[57]–[62]	https://www-nlpir.nist.gov/projects/duc/data.html
DUC 2003	600 sentences	English	[51], [55], [58], [63]–[67]	https://www-nlpir.nist.gov/projects/duc/data.html
DUC 2004	1000 sentences	Arabic, English	[68]–[74]	https://www-nlpir.nist.gov/projects/duc/data.html
DUC 2005	1600 sentences	English	[75]–[78]	https://www-nlpir.nist.gov/projects/duc/data.html
DUC 2006	1250 sentences	English	[77], [79]–[83]	https://www-nlpir.nist.gov/projects/duc/data.html
DUC 2007 [84]	250 sentences	English	[85]–[89]	https://www-nlpir.nist.gov/projects/duc/data.html
EASC	153 documents	Arabic	[6], [90]–[92]	https://www.lancaster.ac.uk/staff/elhaj/corpora.htm
TeMario corpus	1000 sentences	Brazilian, Portuguese, Spanish	[93]–[97]	https://www.linguateca.pt/Repositorio/TeMario/
Enron email dataset	619,446 messages	English	[32], [33], [98]–[101]	https://github.com/deepmind/rc-data/
CNN/Daily mail	One million news	English	[102]–[107]	https://github.com/abisee/cnn-dailymail
Wikihow [108]	200k	English	[109]	https://wikiHow.com
SUMMAC dataset [110]	800,000 electronic documents	English	[15], [111]–[116]	http://www-nlpir.nist.gov/related_projects/tipster_summac/cmplg-xml.tar.gz
SummBank	400 sentences	Chinese, English	[17], [117]–[119]	https://catalog.ldc.upenn.edu/LDC2003T16
Opinosis [120]	575 sentences	English	[86], [121], [122]	http://kavita-ganesan.com/opinosis-opinion-dataset/
TAC 2008	960 sentences	English	[123]–[128]	https://tac.nist.gov/data/index.html
TAC 2009	880 sentences	Multi-media	[129]–[133]	https://tac.nist.gov/data/index.html
TAC 2010	920 sentences	English	[131], [134]–[138]	https://tac.nist.gov/data/index.html
TAC 2011	880 sentences	English	[92], [131], [135], [139]–[142]	https://tac.nist.gov/data/index.html
2013 TREC	1 billion	English	[143]–[147]	https://trec.nist.gov/data.html
2014 TREC	1 billion	English	[116], [146], [148], [149]	https://trec.nist.gov/data.html
2015 TREC	38GB	English	[150]–[152]	https://trec.nist.gov/data.html
LCSTS [153]	2,400,591	Chinese	[154]–[158]	http://icrc.hitsz.edu.cn/Article/show/139.html
Blog Summarization Dataset	350 sentences	English	[15], [159]–[162]	(http://cosmicvariance.com) (http://blogs.msdn.com/ie/)

summarization datasets have four primary sources: newspapers, articles or blogs, reviews, and emails or messages based on types. Newspaper source datasets are TeMario Corpus, CNN News, and Daily mail dataset; Articles or blogs type datasets are EASC, LCSTS, and Wikihow; Reviews source datasets are New Taiwan Weekly, and Opinosis; Emails or messages source datasets are SKE, and Enron dataset. The most popular datasets in ATS are presented in Table 2.

VI. PRE-PROCESSING TECHNIQUES IN ATS

Several pre-processing are performed to clean the noisy and unfiltered text. Erroneous messages and chats, including slang or trash phrases, are known as “noisy” and “unfiltered text”. The approaches mentioned below appear to be some of the most often utilized pre-processing procedures:

- 1) **Parts Of Speech (POS) Tagging:** The technique of grouping or organizing text words according to speech categories such as nouns, verbs, adverbs, adjectives, etc., is known as speech tagging [163].
- 2) **Stop Word Filtering:** Based on the context, stop words are screened out either before or after textual analysis. A, an, and by are illustrations of stop words that can be analyzed and eliminated from plain text [164].
- 3) **Stemming:** Stemming eliminates inflections and derivative forms to a set of words categorized as primary or root forms. By using linguistic strategies such as affixation, text stemming transforms words to consider different word forms [164].
- 4) **Named Entity Recognition (NER):** Words in the input text are recognized as names of items (i.e., person name, location name, company name, etc.) [165].
- 5) **Tokenization:** Tokenization is a text pre-processing technique that divides text flows into tokens, which can be words, phrases, symbols, or other meaningful pieces. The goal of this technique is to examine the words in a document [166], [167].
- 6) **Capitalization:** Diverse capitalization in different documents can be problematic and thus requires to convert every letter into lowercase letters in a document. All text and document words are then merged into a single feature space using this method [168].
- 7) **Slang and Abbreviation:** Slang and abbreviation are two different types of text anomalies that are addressed in the pre-processing stage. A support vector machine is an acronym [169], a shortened form of a word or phrase made up mainly of the first letter of the terms.
- 8) **Noise Removal:** Most textual data contain many more characters, such as punctuation and special characters. While important punctuation and special characters are required for human interpretation of documents, they can cause problems with classification algorithms [170].

- 9) **Spelling Correction:** Spelling correction is an optional step in the pre-processing process. Typos are common in texts and documents, particularly in online media text datasets (e.g., Twitter) [171].
- 10) **Lemmatization:** The process of changing a word's suffix with a new one or eliminating a word's suffix to obtain the basic word form is known as lemmatization (lemma). Its main application area is natural language processing [172], [173].

VII. FEATURE EXTRACTION IN ATS

Feature extraction is a technique for discovering topic sentences, essential data traits or attributes from the source documents. ATS follows two phases to locate the important sentences in the text: extracting features and text representation approach. This section describes the most often used extraction features and text representation approaches for generating sentences for text summarization.

A. FEATURES

Collecting the essential features is the first phase of the feature extraction process. It is necessary to represent the sentences as vectors or score them to find a vital sentence from a document. Some features are used as attributes to define the text for this task. The most prevalent features for calculating the score of a sentence and indicating the degree to which it belongs to a summary are given below:

- 1) **Term Frequency (TF):** The TF metric is used to determine the importance of terms in a single document [174]. As one of the most fundamental properties of ATS, it is commonly employed to represent a word's weight.
- 2) **Term Frequency-Inverse Sentence Frequency (TF-ISF):** The most relevant feature extraction approach based on the text summarization survey measures the term frequency-inverse sentence frequency amongst the sentences in all documents [175]. The weights, which seem to be reasonable indications for meaningful sentences, are generated using this method. Calculating is a quick and straightforward process.
- 3) **Position Feature:** It is usually considered that the beginning and last sentences would provide more information about the document. Researchers have such a better chance of being included in the summary as a result of this. The feature's binary or regressive score value could be anywhere from [0.1] [176].
- 4) **Length Feature:** A sentence's length can indicate whether it is summary-worthy. In summation, it may be wrong to assume that a sentence is worthy of mention based on its length. Compared to the size of other sentences in the source material, very long and comparatively short sentences are usually not included in the summary [177].
- 5) **Sentence-Sentence Similarity:** The resemblance of the querying sentences to other sentences in the text may

be helpful for summarization. This feature extraction process can be performed in various ways [178].

- 6) **Title Feature (Tif):** Sentences containing terms [179] from the headline may suggest the document's theme and are more likely to be included in the summary.
- 7) **Phrasal Information (PI):** The proportion of phrases is always helpful in summarizing. A collection of phrases P includes adjective phrases (ADJP), noun phrases (NP), prepositions (PPM), and verbal phrases (VP) [180].
- 8) **Title Similarity (TS):** A sentence receives a decent grade if it has the most terms in common with the title. The number of words can determine the title similarity in a sentence that appears in the title, and the total number of words [116].
- 9) **Sentence Position (SP):** This feature determines where a sentence appears in the text. The importance of the sentences is decided by where they appear in the text, whether it is the opening of five sentences in a paragraph [177].
- 10) **Thematic Word (TW):** This feature is associated with domain-specific phrases that frequently appear in a text most likely relevant to the document's topic. The score is calculated by comparing the number of theme words in the phrase to the maximum sum of thematic terms in the sentence [179].
- 11) **Numerical Data (ND):** A statement incorporating numerical data is generally crucial. This is most likely found in the summary of a document. The score is calculated by dividing the numerical data in a sentence by the length of the sentence [116].

B. TEXT REPRESENTATION

The text representation models are now utilized to represent the input documents in a better shape. In NLP, text representation approaches imply translating words into numbers so that computers can comprehend and decode patterns within a language. Generally, these approaches develop a connection between the chosen phrase and the context word from the document. Some popular text presentation methods such as bag-of-words, n-gram, and word embedding are discussed below:

- 1) **N-gram:** N-gram is an ideal approach for multi-language operations because it does not require any linguistic preparation. An n-gram is a collection of words or characters with N components. This model is simple to create, and the text may be represented by a vector, which is usually of a reasonable size. Unigrams, bigrams, trigrams, quad grams, and other n-grams comprise a set of text N-grams [181]–[184], [63], [185]. The n-gram has some limitations, such as the fact that the greater the N, the better the model. However, these results go through a lot of processing, requiring heavy computing power in the RAM. N-grams are also a sparse representation of language as the model is based on the likelihood of terms co-occurring. All words that are not present are given a chance of zero in the training corpus.

- 2) **Bag of Words (BoW):** The most primitive sort of numerical text representation is the bag-of-words model [3]. A phrase, such as a term itself, can be expressed as a bag-of-words vector [65]. In a text document, it is a shortened and simplified rendition of the substance of a sentence. Computer vision, NLP, Bayesian spam filters, document categorization, and information retrieval utilizing machine learning are all areas where the BoW technique is used. [101], [186], [187], and [188] are the papers in which BOW feature extraction approaches are used. The following are some of the issues related to BoW: If the new phrases include new words, the vocabulary will expand, as will the length of the vectors. Furthermore, the vectors would have a significant number of elements.
- 3) **Term Frequency-Inverse Document Frequency (TF-IDF):** IDF measures how important the word is, whereas Phrase Frequent (TF) measures how frequently a term appears in a text. The IDF value is needed because merely computing the TF is not sufficient to comprehend the significance of words. The inverse document frequency (IDF) was developed by K. Sparck Jones [189] as a strategy to use term frequency to reduce the impact of implicitly popular terms in the corpus. Term frequency-inverse document frequency is the name given to the combination of TF and IDF (TF-IDF). However, TF-IDF has several drawbacks: it directly calculates texts' resemblance in the word-count space, which might be slow with large vocabularies. Also, it is presumed that the counts of various terms give independent evidence of similarity. [190]–[192], and [175] are examples where TF-IDF is proposed for the feature extraction approach.
- 4) **Word Embedding:** Word embedding is a type of feature learning. Each word or phrase in a lexicon is mapped to an N-dimensional vector of absolute values. Various word embedding algorithms have been proposed to convert n-grams into comprehensible inputs for machine learning systems. This study focuses on Word2Vec, GloVe, and FastText, three of the most widely used deep learning methods for word embedding [2], [193].
 - **Word2Vec:** Word2Vec [4] is a technique for creating embedding. Skip-gram and common bag of words are two approaches (both utilizing neural networks) to obtain it (CBOW). The CBOW technique uses each word's context as an input and attempts to anticipate the word that corresponds to it. Skip-gram aims to optimize the categorization of a word based on another word in the exact phrase rather than expecting the current word based on context [2]. Several articles focused on Word2Vec and can be seen in [50], [101], [194], [195], [194], [196], [197].
 - **Global Vectors For Word Representation (GloVe):** GloVe [198] is another robust word-embedding approach that has been utilized for text categorization. This method is comparable to the Word2Vec process. Each word is represented by a high-dimensional

vector and trained using the surrounding words over a large corpus. [103], [199]–[201] are the articles in which the GloVe word embedding approach was used.

- **FastText:** Several alternative word-embedding representations disregard the morphology of words [202]. By proposing a new word embedding approach called FastText, the Facebook AI Research team introduced a unique solution to tackle this issue. [203]–[205] are the proposed papers in which the FastText word embedding was used.

This section covers all feature extraction methods. The approaches implemented in ATS over the years are detailed in the following section.

VIII. AUTOMATIC TEXT SUMMARIZATION APPROACHES

Generally, ATS is a complex and time-consuming operation that often lacks better results because computers lack a proper understanding of human language. Researchers have tried to extract better performances and standard classifications for summary texts. Text summarization approaches vary based on the number of input documents, such as single or multiple, objective-wise generic, domain-specific, or query-based, and performance-wise. The following sections cover performance-wise analysis, which is divided into two classes.

A. EXTRACTIVE TEXT SUMMARIZATION

The extractive text summarization method aims to identify words and sentences in a text material and use them effectively to create a summary [22]. This requires the selection of sentences from the original document based on their importance. These important sentences are then used to replicate the essential elements of the text word for word, resulting in a subset of the original document's phrases. The foundation is consisted of three independent tasks [152]:

- 1) **Splitting** the source document into sentences and then create an intermediate representation of the text which highlights the task. Intermediate representation has two main types, such as Indicator representation and topic representation [15]
- 2) **Assigning** scores in each sentence for specifying their importance depending on their performance after the representation creation. Topic representation scores on the topic words the text content. On the other hand, indicator representation scores depend on the features of the sentence
- 3) **Selecting** the highest-scoring sentences to form the summary

In extractive text summarization, two approaches of machine learning are applied - supervised and unsupervised machine learning as shown in Figure 3. The following section contains a brief overview of these subclasses.

- **Supervised Learning Methods:** In supervised learning methods, the first step is to learn how to label documents by training to identify summarized and non-summarized

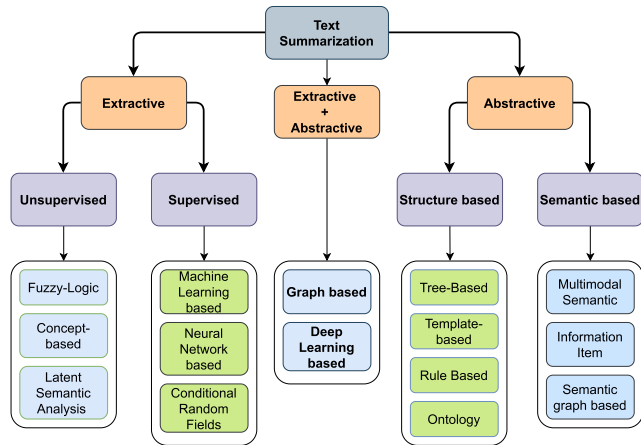


FIGURE 3. An extended classification of Automatic Text Summarization. Text summarization has two approaches and these approaches have two learning methods each whose individual sub-classes are included.

documents. Machine learning and neural network algorithms of these methods require a classified dataset for training, where summarized and non-summarized texts are available with labels [152].

- **Unsupervised Learning Methods:** With unsupervised learning methods, the summarization process can be performed without any help, such as selecting the introductory sentences of the document from the user. These methods only require advanced algorithms such as graph-based, concept-based, fuzzy logic, and latent semantics to take user input and work automatically [152]. These approaches are beneficial for extensive data.

B. ABSTRACTIVE TEXT SUMMARIZATION

Abstractive text summarization is the development and automation of the traditional method of text summarization [147]. The abstractive process identifies key sections and the main ideas of a text document by paraphrasing them. The abstractive summarization process follows some common steps as follows:

- 1) **Analyzing** main contents from the text documents utilizing a vocabulary set different from the source
- 2) **Paraphrasing** the relevant data that fit in the semantics for creating a summary which contains all the actual points of the source document utilizing NLP models

The abstractive summarization approaches are of two types, one is a structure-based approach, and the other one is a semantic-based approach. A brief discussion of these two types based on NLPs is provided below:

- **Structure-Based Methods:** The structure-based approach continuously filters the most critical data from documents by applying abstract or cognitive algorithms. The algorithms for tree-based, template-based ontology, rule-based ontology are the most commonly used [147].
- **Semantic-Based Methods:** The semantic-based approach attempts to refine the sentences by

implementing the NLP on the entire document. This approach can easily find the noun and verb phrases using some methods. These methods are multimodal semantic method (MSM) [206], semantic graph-based method (SGM) [207], information item-based method (IIM) [208], semantic text representation model (STRM) [147], [209].

IX. AUTOMATIC TEXT SUMMARIZATION ALGORITHMS

In this section we explore the supervised, unsupervised, structured-based, semantic-based, extractive and abstractive graph-based and deep learning ATS methods and algorithms.

A. UNSUPERVISED LEARNING METHODS

Unsupervised extractive text summarization picks essential sentences from documents more like regular ATS [210]. This process shows unique nature when it selects sentences without employing labeled summaries during training of the dataset or text document. In addition, unsupervised methods are effective because they do not require user feedback or human overviews to determine the essential features of the document. Unsupervised text summarization is more efficient than supervised and better suited for lengthy text summaries. Various unsupervised techniques or methods for summarizing texts or documents are discussed below:

1) FUZZY LOGIC BASED METHOD

The design of a fuzzy logic-based approach typically includes the selection of fuzzy rules and membership functions. It consisted of four components; a fuzzifier, inference engine, defuzzifier, and knowledge base [211]. The fuzzy-logic-based approach is also used for selecting the most important sentence from the source document or context. However, the fuzzy logic-based method requires a redundancy removal technique to achieve better results. Suanmali *et al.* [212] proposed a fuzzy logic approach for ATS where the summary is created by ordering the ranked sentences from the original text. In addition, some papers such as [213]–[215] focused on this fuzzy logic-based method in their ATS research.

2) CONCEPT-BASED METHOD

The concept-based method extracts concepts and utilizes similarity measures to reduce redundancy from the original document [152]. These retrieved concepts are calculated, and the sentences are scored based on importance. Nevertheless, this method and fuzzy logic have the same limitations, but fuzzy logic stands out as it handles ambiguous situations better. Ramanathan *et al.* [216] proposed a method that employs a sentence concept bipartite graph structure to generate summaries derived from sentences from Wikipedia. More examples for concept-based summarization to retrieve textual concepts from an external information base can be found in [217], [218].

3) LATENT SEMANTIC ANALYSIS (LSA) METHOD

Latent semantic analysis (LSA) is an algebraic-statistical method for extracting hidden semantic structures of sentences and phrases [219]. LSA is an unsupervised learning technique that derives information and similar words from input documents. The significance of this method is that any outside training or template is not necessary to find out similar words appear in separate sentences [220]. However, the LSA method has some limitations, such as not analyzing word order, syntactic relations, or morphologies. In addition, it relies solely on the information contained in the input document rather than on outside knowledge. Finally, limitations such as performance deterioration using inhomogeneous datasets take this method out of comparison. Moreover, this method works quite well in the semantic summary of texts. [42], [54], [221], [222] are other examples of the LSA method for text summarization tasks.

B. SUPERVISED LEARNING METHODS

Supervised learning methods are sentence-level classifications that learn to distinguish between summarized and non-summarized sentences [152]. A collection of documents and human-generated summaries can learn the characteristics of the sentences included in the summary. In addition, it has significant disadvantages in making context summaries manually and requiring more labeled training samples for classification.

1) MACHINE LEARNING (ML) METHOD

The machine learning method is used to classify the sentences as summary or non-summary classes using training data. These methods are applied when multiple document copies require extractive summaries. Notably, in this case, each document's sentences are represented as vectors. Primarily, machine learning algorithms are implemented on a set of trained datasets with documents that are trainable [174]. A collection of training manuals is fed into input documents in the training phase and classified based on the weight of a sentence. In most cases, a simple regression model works better than classifiers but still requires an extensively trained dataset. This data is the only layer in the machine learning algorithm whether a neural network model has multiple layers. That is why neural network models are becoming more usable and user-likelier on ATS. In addition, the ML methods consist of some other standard pre-processing information retrieval algorithms, such as stop-word removal, case folding, and stemming. The stemming algorithm and the concept of sentences represented as vectors are proposed by Porter [223], and [224] respectively. Machine learning trainable algorithms, such as C4.5 or naive Bayes [225] are mostly used where these algorithms are learned on a training set and tested on a separate test set. Several studies have focused on machine learning-based summarization tasks, and it can be seen in [12], [174], [226], [227].

2) NEURAL NETWORK (NN) BASED METHOD

A neural network approach [228] uses a three-layered feed-forward network that learns the features of sentences during model training. The feature matching phase is significant, and the relationship between the characteristics is identified in some steps. Removing infrequent features and combining frequent elements followed by sentence ranking are the steps to define meaningful sentences. The neural network-based method is also used to train as per the human's style or requirements as the network learn from its training data. With multiple layers and increasing the number of hidden layers, NN algorithms perform better than ML algorithms as an advanced version of ML. A framework developed in [229], the RankNet technique also requires neural nets to classify the relevant sentences in the text automatically. It incorporates a two-layer neural network with backpropagation, which is trained using the RankNet algorithm. An old TextSum system architecture including text preparation, keyword extraction, and summary creation was proposed in [230]. The system pre-processes the source document using two methods: stop word removal and stemming. [59], [158], [231], [232], and [176] are other proposed studies in which a neural network was established for summarizing source documents.

3) CONDITIONAL RANDOM FIELDS (CRFs) METHOD

Conditional random fields are statistical modeling techniques based on machine learning that provide a standardized prediction [233]. CRF uses non-negative matrix factorization (NMF) approaches to extract the correct features. Then the proper elements are used to define the introductory sentence from the document. CRF's main benefit is classifying suitable characteristics by offering a more precise representation of sentences and sections. A vital problem of this technique is that it specializes in domain-specific, which necessitates an external domain-specific framework for the training phase. This methodology can be generically employed to any text without first creating a domain framework that is time-consuming. Therefore, ML and NN-based algorithms are still a better option. Some studies have been conducted conditional random field-based methods, and it can be seen in [234]–[236], and [237].

Other methods have been explained in various studies to explain the extractive approach for text summarization. These methods are optimization-based, statistical-based methods, topic-based methods, sentence centrality, or clustering-based methods. Researchers utilized a genetic algorithm to calculate the optimal weights in the optimization method because of its high computational time and cost. The number of iterations required must be defined. The topic-based method concentrates on topics in the input text. The limitation of the topic-based approach is that the sentences will not appear in the summary if the score is not the highest; it affects the quality of the generated summary [238]. Sentence centrality or clustering-based methods includes repeated sentences and is suitable for multi-document summarization. It groups

different sentences on the same topic [239]. However, it requires prior specification of the number of clusters, and for similar sentences, redundancy removal techniques are required [240].

To summarize the text in an abstractive approach, the summary may include a new language that is not seen in the main text, which leads to paraphrasing. Language generation and compression strategies are required to generate abstractive summaries. To generate better abstractive summaries, abstractive text summarization is also divided into two categories, structured and semantic. A brief discussion of structured-based methods, semantic-based methods, and their subclasses is given below:

C. STRUCTURED BASED METHODS

In the abstractive summary, the source document requires newly constructed sentences to summarize. In the structure-based method, phrases from source documents are interpreted in a specified structure without losing their meaning. Structure-based approaches mainly rely on preset forms and spatial reasoning schemas, such as templates, tree-based, ontology-based, and rule-based structures.

1) TREE-BASED METHOD

The tree-based method recognizes sentences that exchange shared knowledge and facts and then mixed them to provide an abstractive summary. This tree-like structure is called tree linearization [241], which comes from many dependency trees. Dependency trees are a representation of the source text of a document. The tree-based model helps process multiple documents and identify the usual information using a syntactic tree. These methods also produce less redundant summaries, but they cannot detect the relationship between sentences without considering the context. Therefore, it overlooks significant phrases in the text. Another issue with this method is the continuous focus on syntax, not semantics. Even after these issues, this model stands out in structured-based methods because of its fluency in summarization.

2) TEMPLATE-BASED METHOD

In the template-based method, the topic or content is extracted into possible phrases and speakers by finding similarities with a template space [242]. The template-based method is used when a document requires a predefined guideline or a human-made template for the summary. This method constructs informative and coherent summaries as various phrases and speakers of the content are selected based on the choice. Oya *et al.* [243] proposed a system that requires a human-made template summary template using a fusion algorithm for multiple sentences.

In another study of Zhang *et al.* [244], a speech act-based strategy was proposed to summarize Twitter topics. The majority of existing Twitter translating algorithms are based on template-based summarization methods. It provides abstract summaries that are appropriate for the many, brief, and chaotic characters of tweets. The issue with this method

is that the templates for summarization are always predefined, which does not give much variety in the summaries. Therefore it cannot produce fluent summaries in comparison to the tree-based approach.

3) RULE-BASED METHOD

The rule-based approach finds facts and reviews of essential concepts in source documents through questioning. The interrogation and questions can be “What are the topic?” “What is the time-being of the story or topic?” etc., and answering these questions tries to generate an abstractive summary. Gupta *et al.* [245] proposed a rule-based method to extract relevant lines from a text paragraph in the Hindi language. Some artificial rules in the Hindi language are employed as “What are person names in the table?” “What are the locations mentioned in the table?” “What are the special symbols contained in the table?” etc. In addition, Laskar *et al.* [246] suggested a method using the BERTSUM model [247], which uses a transformer-based architecture for abstractive summarization. Rule-based methods are used when input documents need to be represented as classes and lists of aspects, such as query-based methods. This method is required to prepare the rules, which is a time-consuming process. Manually written rules make this method less efficient than the other methods mentioned earlier in this subsection.

4) ONTOLOGY-BASED METHOD

Ontology is a knowledge-based approach that acts as a formal naming and definition of the entity types of a specific domain [186]. A base of knowledge is applied in this method to improve the outcome of summarization. Ontology-based methods perform extensively when a document has a knowledge structure or is repeatedly constructed to the same topic. Therefore this method focuses on specific domain-related documents and constructs coherent summaries. Similar to the rule-based method, this method is also time-consuming. Okumura *et al.* [248] proposed a Wordnet ontology in his research work. In other work, Mohan *et al.* [249] proposed some methods for evaluating ontology, such as; ontometric, ontoclean and evalexon. A suitable ontology preparation is a very time-consuming process and cannot be generalized to other domains.

D. SEMANTIC BASED METHODS

Semantic-based methods illustrate the linguistics of a document's texts into a natural language generation (NLG) system, with a significant focus on noun and verb phrase identification [147]. These methods are effective at making less redundant and grammatically correct sentences. A disadvantage of these methods is that they sometimes ignore critical information or data even when grammatically correct.

1) MULTIMODAL SEMANTIC METHOD

The multimodal-based method is used to apprehend both image and text concepts from a document [147]. Therefore

the multimodal semantic model gathers notions and establishes relationships by expressing text and pictures in multimodal materials. The foundation of a semantic model is knowledge representation based on objects. Concepts are represented by nodes, whereas connections represent the relationships between concepts. The completeness, connection with others, and repetitions of an expression are checked using the information density measure. Finally, the selected ideas are translated into sentences to summarize. SimpleNLG is an example of such a system, which provides interfaces for direct control over the way phrases are created and merged and inflectional and morphological control. [250], [251] are examples of multimodal semantic methods utilized in text summarization.

2) SEMANTIC GRAPH BASED METHOD

The semantic graph-based approach summarizes a document by building a graph for the original document called rich semantic graph (RSG) [252] and reducing the created semantic graph. Making brief, cohesive, and grammatically correct sentences with reduced networks is the strength of this method. The semantic-graph-based model mainly extracts semantic information by assigning weights to the nodes and edges of sentences. For this reason, this model works well in most cases but requires a semantic representation of the text. Several studies have proposed semantic graph-based methods for text summarization tasks, and some are [13], [207], [252], [253] and [254].

3) INFORMATION ITEM METHOD

The information-item-based method is used to summarize a text file based on its abstract instead of producing an abstract from the text file's words. An abstractive representation of the source material is used to construct a summary in this approach. The minor component of a source document is an information item [255], [256]. There should be a logical flow of information in a text, and then the method retrieves information from that. A method based on information items delivers more concise and fewer redundant summaries.

E. EXTRACTIVE + ABSTRACTIVE

1) GRAPH BASED METHOD

The graph-based method can be applied to both extractive and abstractive text summarization. This approach is an unsupervised learning method that rates the required sentences or terms using a graph. The purpose of the graphical process is to extract the most relevant sentences from a single text [152].

Graph-based ranking algorithms determine the relevance of a vertex in a graph based on global information iteratively extracted from the entire chart. When it comes to text summarization, specific graph-based techniques are applied.

1) **LexRank:** LexRank is a probabilistic graph-based technique for calculating sentence significance based on the notion of eigenvector centrality in a graph representation

of phrases for natural language processing. It is a connectivity matrix based on intra-sentence cosine similarity that is utilized as the adjacency matrix in a sentence graph representation [65]. [257]–[260], [261] proposed the LexRank algorithm for graph-based text summarization task.

- 2) **Hyperlink-Induced Topic Search (HITS):** Hyperlink-induced topic search is a link analysis algorithm that determines the authority and hub values. The results for the search query are retrieved, and then the computation is performed only on this set of results. A hub value is the total of the scaled authority values of the pages it points to, and an authority value is the sum of the scaled authority values of the pages it points to [262]. Some articles focused on the HITS ranking algorithm for graph-based text summarization tasks, as can be seen in [263]–[265], and [266].
- 3) **PageRank:** The PageRank algorithm utilizes the inbound links of specified pages to measure their significance or quality to rank the search results. PageRank links more weight based on the importance of the page from which it originates [267]. Some articles proposed the PageRank algorithm for summarization [268]–[271], [272].
- 4) **TextRank:** TextRank is an unsupervised method for automatic text summarization to extract the most important keywords from a document. Based on the material that both phrases contain, TextRank estimates the degree of similarity between them [273]. This overlap is computed as the number of shared lexical tokens divided by each sentence's length [274]. [264], [275], [276], and [269] are examples where the TextRank algorithm is proposed.
- 5) **Positional Power Function:** The positional power function is a ranking method that calculates a vertex's score as a function that incorporates both the number and score of its descendants [264]. [277], [278], and [279] are the papers where the positional power function is used.
- 6) **Undirected Graph:** Undirected graphs, in which the out-degree of a vertex is equal to the in-degree of the vertex, can also be used with a recursive graph-based ranking method. Undirected graphs exhibit slower convergence curves for weakly linked graphs with the number of edges proportionate to the number of vertices [264]. Some studies have focused on undirected graph-based algorithms [280], [281], and [282].
- 7) **Weighted Graphs:** Multiple or partial connections between the units (vertices) retrieved from the text may be present in graphs created from natural language texts. As weight is applied to the matching edge that links the two vertices, it may be beneficial to express and include the "strength" of the relationship between two vertices in the model. When calculating the score associated with a vertex in the graph, the edge weights are considered. It is worth noting that integrating vertex weights may be performed using a similar method. [65], [263], [264] are the examples of the articles in which the weighted graph algorithm is used.

8) **Graph-Based Attention Mechanism:** The relationship between all other phrases determines the significance score in the graph model. Traditional attention and graph ranking algorithms are combined in this mechanism to compute the rank scores of the original sentences, resulting in varying significance ratings of actual phrases while decoding various states [283]. Some articles are proposed graph-based attention mechanism for the text summarization task [194], [197].

2) DEEP LEARNING ALGORITHM

Deep learning models help information-driven ATS to become more efficient, accessible, and user-friendly. These models are highly promising for ATS because they attempt to imitate human brain functions. Deep neural networks are commonly employed in NLP issues because their design fits well with the language's complicated structure; for example, each layer can handle a particular job before passing the output to the next. A few commonly known deep-learning models [284] for ATS are described below:

- 1) **RNN Encoder-Decoder:** The sequence-to-sequence paradigm is used in the RNN encoder-decoder architecture. The sequence-to-sequence model converts the input sequence of the neural network into an identical series of letters, words, or sentences. Machine translation and text summarization are two examples of NLP applications [285]. The challenge behind this RNN seq2seq is that it requires an extensive dataset. The training process of datasets is time-consuming. This is why the deep learning methods mentioned in the later part perform better. Anyway, there are some other papers that proposed RNN encoder-decoder in the text summarization task, and some of them are [50], [286], [287], and [155].
- 2) **Long Short-Term Memory (LSTM):** The repeating unit of the LSTM architecture comprises input/read, memory/update, forget, and output gates [6], [288]. The chain structure is very similar to that of an RNN. The input gate is a randomly initialized vector. The input of the current step is the output of the previous step in future stages. The forget gate is a single-layer neural network with a sigmoid activation function. The sigmoid function's result determines whether the prior state's information should be ignored or remembered. The memory gate controls the influence of recognized information on new information. The output gate controls the quantity of new information transmitted to the next LSTM unit. The LSTM shows promise in producing a concise abstractive summary. [5], [289], [290] used the LSTM-based method to summarize risk.
- 3) **Gated Recurrent Unit (GRU):** GRU is a simplified LSTM with two gates: a reset gate and an update gate with no explicit memory. When all the reset gate elements approach zero, the previously hidden state information is discarded. Only the input vector influences the candidate hidden state. The update gate serves as a forget gate in this situation. LSTM contains a memory unit that offers more control, but the calculation time of the GRU consistently decreases. Furthermore, LSTM makes it easier to modify the parameters of whether the GRU takes less time to train [5]. [153], [155], [291] are studies where the writers focused on the GRU-based method for summarization tasks.
- 4) **Restricted Boltzmann Machine (RBM):** A random-probability-distributed neural network (RBM) is a neural network with random probability distributions. A visible layer of visible neurons (input nodes) and hidden layers of hidden neurons constitute the network (hidden nodes). Every hidden node is connected to every input node in a bidirectional manner. Every hidden node is connected to the bias node. In the visible layer, the input nodes are not linked. In addition, hidden nodes are not connected at the hidden levels [292]. The network is known as a restricted Boltzmann machine because of its limited connections. [293]–[296], [297], [298] are the research that focused on RBM method for text summarization.
- 5) **Naive Bayesian Classification:** The naive Bayesian classification method is used to extract the essential keywords from the text [29]. The Bayes technique is a machine learning approach for estimating differentiating keyword characteristics in a text and retrieving the keyword from the input using this data. The use of this naive Bayesian, score, and timestamp idea together improves the accuracy of summarization. [174], [226], [299] focused on the naive Bayesian classification method for text summarization.
- 6) **Query Based:** The score of sentences in a given document is based on the frequency counts of words or phrases in query-based text summarization [116]. Sentences containing query phrases received higher ratings than sentences containing single query terms. The sentences with the highest scores and their structural contexts are then extracted for the output summary [79], [300], [301] are focused on query-based methods for text summarization.
- 7) **Generic Summarization:** Generic summaries aimed at summarizing the document's significant points [302]. A number of excellent general summary examines the papers' key points focused on generic methods for text summarization. Rather than repeating the same information we provide the references here [221], [222], [303], and [191].
- 8) **Q-Network** Q-network is used to approximate optimal action-value function that measures the action's long-term reward for the agent. Based on a partial summary (current state) and a candidate sentence, the model may generate a Q-value (action) [304]. When the agent picks the candidate sentence as part of the summary, the output Q-value reflects the expected value. [305]–[307] proposed a Q-network for text summarization.

Besides these popular models, some pre-trained language models such as BERT, GPT-2, TransformerXL, XLnet have improved in many NLP tasks [308], [309], ranging from sentiment analysis to question answering, natural language inference [310], named entity recognition, textual similarity and paraphrasing [311]. These language models are pre-trained on vast amounts of text data and fine-tuned with various task-specific objectives. With an unsupervised goal of masked language modeling and next-sentence prediction, these models can be so helpful. In most situations, pre-trained language models are encoders for natural language comprehension issues, including classification tasks at the sentence and paragraph level [247].

- 1) **BERT:** Bidirectional Encoder Representations from Transformers (BERT) [312] is a simple and powerful pre-trained model. BERT is developed to alter both left and right context in all layers to pre-train deep bidirectional representations from the unlabeled text. Also, only one output layer of BERT provides state-of-the-art models for various tasks, such as language inference and question answering. It does not require notable task-specific architectural modifications during fine-tuning. BERT uses a Transformer mechanism that learns contextual relations between words in a text. It also includes two separate tools: an encoder that reads the text input and a decoder that predicts. Unlike directional models that read the text input sequentially (left-to-right or right-to-left), the Transformer encoder reads the entire sequence of words at once [313]. Therefore, it is regarded as bidirectional and, in some cases, non-directional to be more accurate. This capacity to integrate both sides significantly aids BERT in achieving better results.
- 2) **GPT2:** The OpenAI GPT-2 [314] has manifested a remarkable capability to formulate coherent and robust summaries than the current language models. The GPT-2 is not a unique design but quite identical to the decoder-only transformer. The GPT2 is an extensive, transformer-based language model trained on a massive dataset called Webtext. The critical difference between GPT2 with BERT is that GPT-2 is built using transformer decoder blocks whether BERT uses transformer encoder blocks. The GPT2 and some later models like TransformerXL and XLNet are auto-regressive. The idea of autoregression is that whenever the token is produced and added to a sentence sequence, the new line becomes the input to the model in its next step [315].

X. SUMMARY OF PAPERS REGARDING ATS

Recently, researchers have been interested in the ATS domain, which has pushed this area to become an excellent exploration topic. In addition, the desire for a better text summarization method has received special attention. This section first presents the significant research on ATS and then

explains the sub-domains of ATS, which are presented in Tables 3, 4, 5, and, 6 with respective state-of-the-art accuracy.

Zhang *et al.* [63] proposed a sentence similarity computation approach using free DUC 2003 datasets and evaluated them with ROUGE-1, ROUGE-2, etc. Mitra *et al.* [317] explored the extraction of two humans for the same paragraph in four quantities: optimistic evaluation, pessimistic evaluation, intersection, union, which performs comparably better than a random selection of sections. Pal *et al.* [318] proposed a system with an online semantic dictionary such as the WordNet and Lesk algorithm. The proposed methodology achieved the best outcomes up to a 50% summary of the original text. Fattah *et al.* [176] explored the use of GA, MR, FFNN, PNN, and GMM for ATS was applied to Arabic and English articles. Ryang *et al.* [319] suggested a framework constructing a summary within the context of reinforcement learning regarding ROUGE scores performed on DUC 2004. Nenkova *et al.* [55] attempted to compare the classifications of baselines on the DUC dataset, while the most challenging task was to provide a focused overview in response to a question/topic. Jing [301] created a unique sentence-reducing algorithm that removes unnecessary phrases and made intelligent reduction judgments based on syntactic information, context, and probability derived from content analysis. Silber *et al.* [320] provided a linear-time technique for lexical chain computing for ATS utilizing 24 documents, including human-generated summaries. Chua *et al.* [321] created a framework using the decay topic model (DTM). The Gaussian decay topic model (GDTM) experimented on Wikipedia links, with GDTM having the best overall performance. Sankarasubramaniam *et al.* [216] presented an analysis of multi-document summarization using TAC 2010. With the assistance of a closely connected paragraph, the performance of the system is significantly improved.

However, the papers of the four famous sub-domains of ATS are now explained in the following subsections.

A. SUMMARY OF THE PAPERS REGARDING UNSUPERVISED LEARNING METHOD

Previously we mentioned that the extractive text summarization technique consists of two learning methods. In this section, we discuss the papers that cover the unsupervised learning methods of ATS.

Steinberger *et al.* [191] used latent semantic analysis (LSA) to locate semantically significant sentences using two different evaluation approaches. Suanmali *et al.* [212] proposed a fuzzy logic relatable sentence extraction summarizer using the DUC 2002 dataset where a sentence of the document was extracted and expressed as a vector of characteristics. Erkan *et al.* [65] provided a way to determine graph-based sentence centrality scoring suggesting a framework comprises three distinct approaches for calculating centrality in similarity graphs. Yeh *et al.* [42] provided two new modified methods for ATS with the corpus-based approach (MCBA) and the LSA-based TRM approach

TABLE 3. A set of papers on unsupervised learning methods are summarized in this table.

Task	Method	Dataset	Feature Extract	Accuracy	Limitation & Future Work
Text summarization by sentence selection [212]	Fuzzy logic	DUC 2002	Tif, SL, TW, SP, TW, etc.	F-measure: 0.498, recall: 0.457 and precision: 0.471 respectively	<ul style="list-style-type: none"> - Fuzzy logic did not produce promising results - Authors would like to use other learning with fuzzy logic in their future work
Sentence similarity measure [101]	TF-IDF, Word2vec, Sentence2vec	SKE for English and Arabic	AE, VAE, ELM-AE	ROUGE-1: 0.6043 and ROUGE-2: 0.5771	<ul style="list-style-type: none"> - Arabic abstractive summarization was not possible - Authors would like to use auto encoders or attention encoders in their future work
Sequence labeling [235]	Conditional Random Fields (CRF)	DUC 2001	SP, SL, Log-likelihood, TW, Indicator words etc.	ROUGE-2 :0.245, F1 Measure: 0.202 of	<ul style="list-style-type: none"> - Linguistic features were not used - Authors would like to use rhetorical structures for robustness in their future work
Short representation from a long document [316]	Latent Semantic Analysis (LSA)	Corpus of Contemporary Arabic (CCA)	Euclidean Distance, Cosine Similarity, Jaccard Coefficient etc.	Entropy 0.1275(approx.), Purity 0.385(approx.) and	<ul style="list-style-type: none"> - NA
Query-based single-document summarization [99]	Restricted Boltzmann Machine (RBM)	SKE, BC3, TAC	Key phrase oriented, Subject oriented summarization	Precision: 0.1694, Recall: 0.2115 and F-score of AE: 0.1816	<ul style="list-style-type: none"> - Did not cover multi-documents summarization into this system - Authors would like to extend generic summarization by clustering

(LSA + TRM) [328] and a text connection map to extract semantically significant structures from a document.

Alami *et al.* [101] used neural network-based techniques for ATS using the Sentence2Vec feature extraction approach, which produced the best outcomes. Gong *et al.* [222] proposed two text summary approaches for creating a general text summary by minimizing redundancy. Shen *et al.* [235] solved a sequence labeling problem by employing the effective sequence-labeling algorithm CRF. Froud *et al.* [316] suggested enhancing the functionality of summarization using the latent semantic analysis model and Arabic document clustering measures with stemming. Mihalcea *et al.* [264] studied and evaluated a variety of graph-based ranking algorithms that enable automatic unsupervised sentence extraction from the perspective of ATS. Yousefi *et al.* [99] introduced an unsupervised deep neural network using the SKE and BC3 email datasets that employ global and local vocabularies to represent words as the AE input.

B. SUMMARY OF PAPERS REGARDING SUPERVISED LEARNING METHOD

In this section, we discuss the papers that covers the supervised learning methods of ATS.

Xu *et al.* [327] introduced a neural network architecture for extractive summarization, consisting of a sentence extraction model and a compression classifier. According to the results of Liu *et al.* [37], constructing English Wikipedia articles can be addressed as a multi-document extractive summarization of original documents with a decoder-only sequence transduction architecture. Xu *et al.* [322] introduced DISCOBERT, which employed discourse units as the lowest selection basis to eliminate summarization redundancy and utilizes two types of discourse graphs. Alguliyev *et al.* [324] suggested an unsupervised approach with sentence clustering using the open datasets DUC 2001 and 2002. Aliguliyev *et al.* [323] also described a strategy for sentence clustering using a discrete differential evolution technique. In addition, the NGD-based dissimilarity measure outperformed Euclidean distance.

Ferreira *et al.* [325] evaluated 15 sentence scoring methods on three distinct datasets (news, blogs, and article settings) to improve the acquired sentence extraction findings. Neto *et al.* [174] investigated the framework using an ML method by utilizing statistics-oriented techniques where the Naive Bayes method and the C4.5 decision tree method are the best classification methods. Fang *et al.* [326] investigated

TABLE 4. A set of papers on supervised learning methods are summarized in this table.

Task	Method	Dataset	Feature Extract	Accuracy	Limitation & Future Work
Two-stage extractive and abstractive framework [37]	Highlighting the relevant information	WikiSum (self-generated)	TF-IDF, TextRank, SumBasic, T-D, T-DMACA etc.	ROUGE-L: 38.8	<ul style="list-style-type: none"> - Required performance measurement improvement - Authors would like to conduct further research to improve performance
Decrease summarization redundancy with DISCOBERT [322]	BanditSum, NeuSum, JECS, BertSum, PNBert, HiBert	New York Times (NYT), CNN, and Daily mail (CNNDM)	EDU	ROUGE-1: 50.00, ROUGE-2: 30.38 and ROUGE-L: 42.70	<ul style="list-style-type: none"> - Did not apply different task on long document encoding - Authors would like to explore better encoding methods in their further research
To find Solution to information overload [323]	Sentence clustering	DUC 2001, 2002	Stop words, Stemming	ROUGE-1: 0.47856, ROUGE-2: 0.18528 and F1 measure: 0.48324	<ul style="list-style-type: none"> - Author tested one method only - Authors would like to explore more datasets to evaluate the model
Enhancing criterion and objective functions [324]	Discrete differential evolution algorithm	DUC 2001, 2002	Sentence similarity, Sentence clustering	ROUGE-1: 0.45412, ROUGE-2: 0.18982	- NA
Quantitative and qualitative assessment of 15 algorithms [325]	Word scoring, Sentence scoring and graph scoring	CNN Dataset, Blog summarization dataset, SUMMAC dataset	Word Frequency, TF/IDF, Lexical Similarity, SL, SP, Text rank etc.	Average Recall: 0.47, Precision: 0.19, and F-measure is: 0.26	- Authors would like to compare with other algorithms in their future work
[Do not touch I am working from 6.30pm again] Word-sentence co-ranking model [326]	CoRank model	DUC 2002	Redundancy removal strategies	F-measure: 0.59, ROUGE-1: 0.69 and ROUGE-2: 0.60	<ul style="list-style-type: none"> - Did not conduct query-oriented and event-driven summarization tasks - Authors would like to extend co-rank model multi-document summarization
Joint Extractive and Compressive Summarizer (JECS) [327]	Neural Network model	CNN/Daily Mail, New York Times	Sentence and Document Encoder, Compression Encoder, Compression Classifier	ROUGE-1: 45.5, ROUGE-2: 25.3	- NA

a graph-based ranking model using redundancy removal strategies to enhance the effectiveness of the summarization process. Kaikhah *et al.* [228] described artificial neural networks to generate summaries of news stories of different lengths using feature fusion to summarize highly ranked sentences. Ledeneva *et al.* [331] provided a statistical approach for single-document extractive ATS that generates a text summary by extracting selected sentences from the source.

C. SUMMARY OF PAPERS REGARDING STRUCTURED TEXT SUMMARIZATION

Structured-based methods are a vital part of abstractive text summarization approaches. The following section examines studies that discuss structured learning methods.

Li *et al.* [155] designed a methodology for ATS based on a seq2seq encoder-decoder architecture with a deep recurrent generative decoder (DRGN). Liu *et al.* [330] proposed an adversarial technique for ATS that trained both a generative model and a discriminative model simultaneously. Hennig *et al.* [186] described how sentences can be mapped to nodes with several linguistic features that are generated to test the efficiency of an SVM classifier. Genest *et al.* [256]

presented a methodology for information extraction and natural language generation. Kikuchi *et al.* [329] developed an approach for summarizing a single text that contained both sentence and word relationships in a hierarchical tree and the ROUGE score compared to EDU selection. Song *et al.* [5] constructed a novel ATSDL system based on an LSTM-CNN that solves numerous challenges in text summarizing. Oya *et al.* [243] demonstrated an ATS for meeting discussions based on modifying a word graph algorithm to build frameworks from human-generated summaries.

D. SUMMARY OF PAPERS REGARDING SEMANTIC TEXT SUMMARIZATION

Based on a comprehensive review of structured learning methods, the following section focuses only on semantic learning approaches for ATS.

Wang *et al.* [154] introduced a joint attention and biased probability generation approach using three datasets, DUC 2004, Gigaword, LCSTS, where ConvS2S architecture improved by topic embedding, and SCST provided the best results. Chen *et al.* [332] presented a unique sentence-level policy gradient strategy between two neural networks

TABLE 5. A set of papers on structured based learning methods are summarized in this table.

Task	Method	Dataset	Feature Extract	Accuracy	Limitation & Future Work
Condensed version by selecting informative sentences [186]	SVM classifier	DUC 2002	Stop words, Stem and cosine similarity	ROUGE-1 F1: 0.5716, ROUGE-2 F2: 0.3143	<ul style="list-style-type: none"> - Did not cover extensive ontology for further effectiveness - Authors would like to explore the approach in non-hierarchical ontology
Fully abstractive approach [256]	Information Extraction, Content Selection, Generation	TAC 2010 and 2011	Thesaurus to locate semantically related nouns and verbs	Pyramid score: 0.54	<ul style="list-style-type: none"> - Did not include expanded variety for other categories and more aspects - Authors would like to compare proposed model in a greater scale
Sentence level dependency to create nested parent and child trees [329]	Dependency-based discourse trees (DEPDTs)	RST-DTB	EDU and ILP formulation	ROUGE-1: 0.354	<ul style="list-style-type: none"> - Only used ROUGE evaluation metrics and rhetorical structures on sentences - Authors would like to utilize individual sentence structures with EDUs
Aim to develop a semantic model [5]	LSTM-CNN model	CNN dataset, DailyMail	MOSP	ROUGE-1: 34.9, ROUGE-2: 17.8	- NA
Performing a variational inference and generation with NN [155]	Seq2Seq with DRGN	DUC2004, Gigaword, LCSTS	DRGD outperforms state-of-the-art approaches	ROUGE-1: 36.71, ROUGE-2: 24.00 and ROUGE-L: 34.10	- NA
An adversarial technique for abstractive text summarization [330]	Generative Model, Discriminative Model	CNN/ Daily Mail	Pre-trained generator and discriminator of human-generated positive examples	ROUGE-1: 39.92, ROUGE-2: 17.65 and ROUGE-L: 36.71	- NA

hierarchically while preserving language proficiency using CNN/daily mail dataset. Kryściński *et al.* [46] developed a method for validating abstractive neural models to perform factual consistency testing on the document-sentence level. Zhu *et al.* [206] proposed a multimodal objective function to utilize the loss through summary generation. ROUGE and order ranking is used to produce the multimodal reference for both automatic and human performance measures. Khan *et al.* [254] proposed the Sem-Graph-Both-Rel method, which was compared to other summarization techniques based on three pyramid evaluation metrics. A strategy for creating an abstractive summary for a single document is described in this paper [252], which uses a rich semantic graph reducing methodology that can reduce the actual document to 50%. Genest *et al.* [255] offered an optimistic abstractive summarization, which aimed at achieving an accurate objective by managing the content and structure of the summary using TAC the 2010s dataset.

XI. PERFORMANCE OF AUTOMATIC TEXT SUMMARIZATION

The evaluation of text summarization is difficult. This task is complex for machines to identify key phrases or contents that are important and add value in summary. Placing key

phrases has changed the meaning of the summary depending on the purpose of the context, and it is challenging to locate this relevant information. As a result, automatic evaluation measures are necessary for reliable and effective evaluation. After reviewing previously researched papers covering text summarization topics, several methods are determined for summarization measurement. Now, the evaluation measurement metrics of the ATS domain are discussed below:

A. EXTRINSIC EVALUATION

Extrinsic evaluation determines the quality of ATS generated summary depending on how it influences other activities such as text categorization, information retrieval, and question responding. In the summarising process, it is considered good if it aids these mentioned other activities. There are numerous approaches to extrinsic evaluation. Relevance assessment determines whether the text is relevant to the topic, and reading comprehension determines whether it can answer multiple-choice assessments or not.

B. INTRINSIC EVALUATION

The intrinsic evaluation determines the quality of the summary based on comparability among the machine-generated

TABLE 6. A set of papers on semantic based learning methods are summarized in this table.

Task	Method	Dataset	Feature Extract	Accuracy	Limitation & Future Work
Incorporated topic information [154]	Convolutional sequence-to-sequence	DUC 2004, Gigaword, LCSTS	Topic embeddings, Joint attention, Biased Probability Generation	ROUGE-1 F, ROUGE-2 F and ROUGE-L F is 36.92, 18.29 and 34.58	<ul style="list-style-type: none"> - Only provides results based on sentence summarization - Authors would like to work further on long paragraphs or multi-document summarization
Verify real consistency and identify conflicts [46]	Abstractive neural model	CNN/ Daily Mail	Entity swap, Pronoun swap, Noise injection etc.	Accuracy: 74.15, F1: 0.5106	<ul style="list-style-type: none"> - Did not cover multi-sentence spans from original documents - Authors would like to expand the proposed model with data augmentation techniques
Condensed version by maintaining significant information [254]	Semantic graph approach	DUC 2002	MMR to Control Redundancy	ROUGE-1: 0.418 0.102	<ul style="list-style-type: none"> - Did not cover hierarchy concept and automated generative abstract summary - Authors would like to explore automatic concept hierarchy learning from a text corpus
Abstractive summarization with proposed method [252]	Rich Semantic Graph	WordNet ontology	Cross-reference resolution, and Pronominal resolution processes	Reduced the actual document to 50%	<ul style="list-style-type: none"> - Proposed approach creates abstract summary only for single-document - Authors would like to develop a prototype to assess the results with multiple documents
Abstract representation of source text [255]	Information Item Retrieval method	TAC 2010	Noun Phrase, Verb phrase	Best Pyramid Score: 0.425	<ul style="list-style-type: none"> - Proposed approach attempted on a single dataset - Authors would like to re-search more to find words and phrases of similar meanings in summarization

and human-generated summaries. A good summary is judged based on two significant factors: quality and information. Human experts may be required to evaluate machine-generated summaries utilizing several quality measures. Readability, non-redundancy, structure, and coherence, and some other quality metrics include referential clarity, conciseness and focus, and content coverage, etc.

Some valuable measures for intrinsically evaluating summaries are precision, recall, and F-measure. Researchers must anticipate comparability between human-generated and automatically generated summaries. With the evaluation metrics mentioned earlier, it is also possible that the two summaries produce different evaluation outcomes despite being equally good. The following section focuses on the most frequently used evaluation metrics in the research, including the following:

1 **Precision Metric:** The precision metric evaluates whether the percentage of sentences chosen by humans and the computer is correct. The formula shows how the precision metric is calculated by dividing the total number of sentences between two summaries by the number of sentences in the system

summary [152], [333], [334].

$$\text{Precision} = \frac{S_{\text{ref}} \cap S_{\text{cand}}}{S_{\text{cand}}} \quad (1)$$

2 **Recall Metric:** The recall metric determines the system recognizes how many sentences are selected by humans. The following equation is calculated by dividing the number of sentences in both the reference and system summaries [152]. These studies used recall metrics for the evaluation measurement task [99], [179], [226], [335].

$$\text{Recall} = \frac{S_{\text{ref}} \cap S_{\text{cand}}}{S_{\text{ref}}} \quad (2)$$

3 **F-Measure Metric:** The F-measure metric incorporates recall and precision metrics. The arithmetic mean of precision and recall is an F-measure metric [152], [42], [212], [336], and [337] focused on the F-measure metric for the evaluation task.

$$\text{F-Measure} = \frac{2 (\text{Precision}) (\text{Recall})}{\text{Precision} + \text{Recall}} \quad (3)$$

4 **ROUGE Metric:** Recall-Oriented Understudy for Gisting Evaluation (ROUGE) is a series of evaluations

ATS and machine translation. It compares an automatically generated summary or translation to a set of predetermined summaries such as human-generated summaries. ROUGE consists of five measures: ROUGE-N, ROUGE-L, ROUGE-W, ROUGE-S, and ROUGE-SU. The examples where the ROUGH metrics were used can be found in [146], [338], [339] and [340].

- **ROUGE-N (R1):** ROUGE-N is focused on the uni-gram measure of a ATS summary against a human-generated or pre-defined reference summary. N-gram recall algorithm that compares the system and reference summaries [16], [341].
 - **ROUGE-L (R-L):** The ROUGE-L process is based on the longest common sub-sequences (LCS) between human-generated and automatic-generated summaries. It evaluates the ratio of the size of the LCS of two summaries to the size of the reference summary [16], [341].
 - **ROUGE-W:** ROUGE-W determines the weighted longest standard sub-sequence, which is an enhancement of the LCS [16].
 - **ROUGE-S:** ROUGE-S (Skip-Bi-gram co-occurrence statistics) measures the percentage of skip bigrams shared between the system and reference summaries. The skip bigrams would be any word pair in the sentence sequence with random gaps [16], [341].
 - **ROUGE-SU*:** ROUGE-SU is extended by employing skip-bi-grams and a uni-gram as a measuring unit, a weighted average of ROUGE-S and ROUGE-N. These metrics allow bi-grams to be made up of non-adjacent words with a maximum of n-words between them [16].
- 5 **Pyramid Method:** The pyramid technique is used because there is no best comparison summary among the human-created model summaries. The fundamental aim is to generate a global standard summary by comparing human-generated comparison summaries based on summary content units (SCUs). A good summary has more SCUs from higher pyramid levels than lower levels, whereas a poor summary has more SCUs from lower tiers than higher tiers [16].
- 6 **Relative Utility:** This measurement assigns a score between 0 and 10 to each sentence in the input document based on relevance. The highest-scored sentence is thought to be more appropriate for summary [342].
- 7 **Basic Elements:** Basic element is a modifier or an argument and the connection of the modifier to the head. The goal of this strategy is to match distinct comparable expressions more easily [16].
- 8 **Text Grammars:** This strategy aids in the evaluation of text summaries. This focuses on identifying the structure of acceptable text in a formalized setting [16].
- 9 **Factoid Score:** Factoid score is the evaluation of computerized summaries in terms of factoids which are atomic units of information. Different pre-defined

summaries are utilized, and shared knowledge is evaluated among these [343].

- 10 **Cohesion and Coherence:** Cohesion attempts to account for relationships between text elements. The four significant forms of cohesion revealed are reference, ellipsis, conjunction, and lexical coherence [344]. And coherence refers to the text's overall unity or cohesiveness, which is accomplished by efficiently grouping and logically arranging ideas. It's expressed in terms of text-to-text relationships, such as elaboration, cause, and explanation. Mani *et al.* [345] addressed the cohesion and coherence in their text summarization task.
- 11 **BLEU:** The Bilingual Evaluation Understudy (BLEU) evaluation metrics assess the output quality of machine translation systems in terms of reference translation [346]. Counting the number of n-gram matches located independently between the system and the reference translations is the main task of this metric [347]. The BLEU metrics can be computed as:

$$BLEU = \min\left(1, \frac{\text{output length}}{\text{reference length}}\right) \left(\prod_{i=1}^4 \text{precision}_i\right)^{\frac{1}{4}} \quad (4)$$

- 12 **CHRF:** Character n-gram F-score (CHRF) generates a simple F-score by combining the recall and precision of character n-grams of maximum length 6 with several parameter values β ($= 1, 2$, or 3) [348]. However, this is not a common evaluation metrics various variants [349], [350] of CHRF are used to measure performance of text representation techniques like word embeddings. The CHRF can be computed as:

$$CHRF = \left(1 + \beta^2 \frac{ChrP \cdot ChrR}{\beta^2 \cdot ChrP + ChrR}\right) \quad (5)$$

Automatic summary evaluation based on n-gram graphs (AutoSummENG), Qarla, ParaEval, GEMS, HowNet, DEPEVAL used automatic evaluation methods. These evaluation measures do not require human annotations, while others are semi-automated such as factoid score, relative utility, pyramid method, and text grammars require some human annotations [185]. In evaluating the text summarization task, there are some issues with the performance measures, which creates challenges. For this reason, researchers could not reach a proper conclusion and generate a promising result. These issues are addressed elaborately in the Section XII.

XII. LIMITATIONS AND CHALLENGES

ATS aims at assisting users in condensing all important information that needs to be summarized. As we have discussed in previous sections, although many summarization techniques can be used to create summaries from texts and documents, these techniques are still confined to extracting specific parts of the original text and concatenating them into a shorter text, abstraction, or paraphrasing the original material in a broad sense. The ultimate goal of any ATS

system should be able to summarize texts as close as possible to a human-generated summary. However, to reach this goal, existing ATS systems still have significant important challenges. Some common challenges include ‘anaphora problem’ and ‘cataphora problem’ lag etc. In this section, we explore and investigate the most common challenges of the ATS domain.

The challenges of ATS tasks for both extractive and abstractive summarization techniques are:

- 1) **Evaluation:** This is a commonly encountered difficulty in automatic text summarization. The same study with a different dataset and metrics evaluated different types of results. Datasets can be biased to some techniques, such as datasets and metrics biased toward extractive summarization. Then, using a common dataset and metric can produce a good result. However, automatic evaluation techniques have several issues that should be addressed. The analysis of the summarization task with precision and recall may be deceiving to the researchers and do not correspond to the desired conclusion. Considering, precision and recall do not put the knowledge of the source documents at stake while comparing. The agreement, which takes two experts to achieve randomly, is determined by the number and percentages of the classes that the researchers perform [351]. In the case of an extractive summary, the score generated with automated evaluation techniques such as ROUGE, BLEU, etc., shows less significance than the human evaluation score. Due to losing sensitivity, some performance measures also fail to generate scores on higher-quality summarization. These techniques also lack diversity in vocabulary that creates indecision to find synonyms of words used in documents. In addition, automatic evaluation techniques require reference sentences, and collecting a vast amount of reference sentences is a difficult task. Lastly, semantically and syntactically incorrect sentences are ignored while scoring. It is a significant issue as some of these metrics give good scores to trivial sentences and fail to evaluate grammatically incorrect sentences [352], [353].
- 2) **Important Sentence Selection:** Usually, an ATS system selects the most relevant sentences from the original text and marks them essential. While forming the summary, selective sentences or words need to be standard as per the benchmark. However, giving significance to the sentences is very subjective. Thus, while making any summary from particular sentences, it would make a difference. This problem can be solved by using user-specific data to work in a professional summarise to produce summaries. Although vector representation and similarity matrices attempt to find word correlations, there is no reliable way to identify the most important sentences.
- 3) **Lack of Different Scenario-Based Training Data:** There is much information on the Internet with various scenarios, such as politics, crickets, and statistical data representation. The datasets used in the summarization tasks did not cover every different method for training.
- 4) **Interpretability:** Abstractive models provide condensed representations of the source content that express its essential concepts. Machines struggle with the complexities of human language and how humans express emotions, particularly in written materials. Therefore, ensuring the interpretability of source content through abstract models is a difficult task.
- 5) **Interpreting Long Sentences and Jargons:** Most of the existing learning methods can only summarize short sentences and get puzzled when long sentences are encountered by the algorithms while processing the source text. Researchers should identify the problem and thus build a new architecture to reduce or eliminate this problem to solve this issue.
- 6) **Anaphora Problem:** Anaphora problem is a prevalent difficulty in text summarization. During the discussion, humans, frequently substitute the subject with synonyms or pronouns. The ‘anaphora problem’ determines which pronoun complements which word.
- 7) **Retaining the Quality of the Text:** An ATS should ensure the quality of the summarized text. From the user perspective, the most desired quality of ATS is to understand the source text while summarizing. Various machine learning techniques can be used to retain the quality of the summarized text.
- 8) **Word Sense Ambiguity:** Ambiguity in words makes a difference while summarizing sentences. This ambiguity may appear due to abbreviations with more than one acronym, multiple usages of the same word in different contexts, etc. Then the acronym has to match the topic or meet the sense depending on the subject for better understanding. This problem is the opposite of the anaphora problem, which is called the Cataphora problem. Therefore, this problem can be solved using a disambiguation algorithm.
- 9) **Meaningful, Intuitive, and Robust:** Summarized sentences must be influential or make sense to the users, and representation must be strong concerning any areas the system faces.
- 10) **Predefined Template:** Recently, natural language processing has made an incredible amount of progress in ATS. But these methods cannot generate new sentences on their own. Therefore, the template-based algorithm was introduced, where a specific template needs to be predefined for a particular summarization task.
- 11) **Attaining Higher Level of Abstraction:** In a text summarization task, an open research topic is the achievement of a higher-level abstraction. Therefore, there are plenty of possibilities for researchers and linguistics to find the answer to this problem.

In addition to the above-mentioned general challenges, we also present a few limitations of the current algorithms used in the ATS domain. Table 7 presents the limitations those

TABLE 7. The table presents the limitations of ATS algorithms.

		Algorithm	Limitations
Extractive	Unsupervised	Fuzzy logic	It requires a redundancy removal technique in the post-processing phase to improvise the summarization quality.
		Concept-based	It needs to utilize similarity measures for reducing redundancy which can affect the quality of the summary.
		Latent-Semantic	The LSA generated summary required a large amount of time.
	Supervised	Machine Learning	It needs a large set of data for training and improving the sentence selection for making a good summary.
		Neural Network	It is quite slow in training phase and application phase. Also requires human interruption for training data.
		Conditional Random Fields	In CRF linguistic features are not considered. It also requires external domain specific corpus.
Abstractive	Structure-based	Tree based	It ignores the context and significant phrases in the text, eventually failing to detect the relation between sentences. Another drawback is that it continuously focuses on syntax, not semantics.
		Template based	The templates are pre-defined in this method that creates a lack of diversity in the summaries.
		Rule-based	The requirement to prepare the rules is a time-wasting process. Another challenge is that the rules needed to be written manually.
		Ontology-based	A suitable ontology preparation is a very time-consuming process and cannot be generalized to other domains.
	Semantic-based	Multi-modal semantic	An automatic evaluation of the framework is required as it is manually evaluated by humans.
		Information item	Difficulties of creating meaningful and grammatical sentences from text. Also linguistic quality of summaries is very low due to incorrect parses.
		Semantic graph	This method is limited to single document abstractive summarization.
		Deep learning	It need human effort for building big training data manually.
		Graph-based	It does not consider importance of words and does not consider dangling anaphora problem.

are needed to be solved to achieve better text summarization results.

XIII. CONCLUSION

Text summarization is an old topic, but this field continues to gain the interest of researchers. Nonetheless, the performance of text summarization is average in general, and the summaries created are not always ideal. As a result, researchers are attempting to improve existing text-summarizing methods. In addition, developing novel summarization approaches to produce higher-quality, human standards and robust summaries is a priority. Therefore, ATS should be made more intelligent by combining it with other integrated systems to perform better. Automatic text summarization is an eminent domain of research that is extensively implemented and integrated into diverse applications to summarize and reduce text volume. In this paper, we present a systematic survey of the vast ATS domain in various phases: the fundamental theories with previous research backgrounds, dataset inspections, feature extraction architectures, influential text summarization algorithms, performance measurement matrices, and challenges of current

architectures. This paper also presents the current limitations and challenges of ATS methods and algorithms, which would encourage researchers to try to solve these limitations and overcome new challenges in the ATS domain.

ACKNOWLEDGMENT

The authors would like to thank the Advanced Machine Learning (AML) Laboratory for resource sharing and precious opinions.

REFERENCES

- [1] H. P. Luhn, "The automatic creation of literature abstracts," *IBM J. Res. Develop.*, vol. 2, no. 2, pp. 159–165, Apr. 1958.
- [2] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," 2013, *arXiv:1301.3781*.
- [3] Z. S. Harris, "Distributional structure," *Word*, vol. 10, nos. 2–3, pp. 146–162, 1954.
- [4] K. W. Church, "Word2Vec," *Nat. Lang. Eng.*, vol. 23, no. 1, pp. 155–162, 2017.
- [5] S. Song, H. Huang, and T. Ruan, "Abstractive text summarization using LSTM-CNN based deep learning," *Multimedia Tools Appl.*, vol. 78, no. 1, pp. 857–875, Jan. 2019.
- [6] Q. A. Al-Radaideh and D. Q. Bataineh, "A hybrid approach for Arabic text summarization using domain knowledge and genetic algorithms," *Cognit. Comput.*, vol. 10, no. 4, pp. 651–669, Aug. 2018.

- [7] K. Ježek and J. Steinberger, "Automatic text summarization (the state of the art 2007 and new challenges)," in *Proc. Znalosti*. Princeton, NJ, USA: Citeseer, 2008, pp. 1–12.
- [8] B. Kitchenham and S. Charters, "Guidelines for performing systematic literature reviews in software engineering," Keele Univ., Durham Univ., Durham, U.K., Tech. Rep. EBSE-2007-01, Version 2.3, 2007.
- [9] B. Kitchenham, *Procedures for Performing Systematic Reviews*, vol. 33. Keele, U.K.: Keele Univ., 2004, pp. 1–26.
- [10] S. Gholamrezazadeh, M. A. Salehi, and B. Gholamzadeh, "A comprehensive survey on text summarization systems," in *Proc. 2nd Int. Conf. Comput. Sci. Appl.*, Dec. 2009, pp. 1–6.
- [11] C. Saranyamol and L. Sindhu, "A survey on automatic text summarization," *Int. J. Comput. Sci. Inf. Technol.*, vol. 5, no. 6, pp. 7889–7893, 2014.
- [12] R. Mishra, J. Bian, M. Fiszman, C. R. Weir, S. Jonnalagadda, J. Mostafa, and G. D. Fiola, "Text summarization in the biomedical domain: A systematic review of recent research," *J. Biomed. Informat.*, vol. 52, pp. 457–467, Dec. 2014.
- [13] N. Andhale and L. A. Bewoor, "An overview of text summarization techniques," in *Proc. Int. Conf. Comput. Commun. Control Autom. (ICCUBEA)*, Aug. 2016, pp. 1–7.
- [14] S. K. Bharti and K. S. Babu, "Automatic keyword extraction for text summarization: A survey," 2017, *arXiv:1704.03242*.
- [15] M. Allahyari, S. Pouriyeh, M. Assefi, S. Safaei, E. D. Trippe, J. B. Gutierrez, and K. Kochut, "Text summarization techniques: A brief survey," 2017, *arXiv:1707.02268*.
- [16] M. Gambhir and V. Gupta, "Recent automatic text summarization techniques: A survey," *Artif. Intell. Rev.*, vol. 47, no. 1, pp. 1–66, Jan. 2017.
- [17] W. S. El-Kassas, C. R. Salama, A. A. Rafea, and H. K. Mohamed, "Automatic text summarization: A comprehensive survey," *Expert Syst. Appl.*, vol. 165, Mar. 2021, Art. no. 113679.
- [18] L. Abualigah, M. Q. Bashabshah, H. Alabool, and M. Shehab, "Text summarization: A brief review," in *Recent Advances in NLP, the Case of Arabic Language*. Cham, Switzerland: Springer, 2020, pp. 1–15.
- [19] R. Mihalcea and H. Ceylan, "Explorations in automatic book summarization," in *Proc. 2007 joint Conf. empirical methods natural Lang. Process. Comput. natural Lang. Learn. (EMNLP-CoNLL)*, 2007, pp. 380–389.
- [20] N. V. Kumar and M. J. Reddy, "Factual instance tweet summarization and opinion analysis of sport competition," in *Soft Computing and Signal Processing*. Singapore: Springer, 2019, pp. 153–162.
- [21] P. Gupta, R. Tiwari, and N. Robert, "Sentiment analysis and text summarization of online reviews: A survey," in *Proc. Int. Conf. Commun. Signal Process. (ICCSPP)*, Apr. 2016, pp. 241–245.
- [22] V. Gupta and G. S. Lehal, "A survey of text summarization extractive techniques," *J. Emerg. Technol. Web Intell.*, vol. 2, no. 3, pp. 258–268, 2010.
- [23] S. Muresan, E. Tzoukermann, and J. L. Klavans, "Combining linguistic and machine learning techniques for email summarization," in *Proc. Workshop Comput. Natural Lang. Learn. (ConLL)*, 2001, pp. 1–8.
- [24] S. D. Kavila, V. Puli, G. P. Raju, and R. Bandaru, "An automatic legal document summarization and search using hybrid system," in *Proc. Int. Conf. Frontiers Intell. Comput., Theory Appl. (FICTA)*. Berlin, Germany: Springer, 2013, pp. 229–236.
- [25] H. D. Menendez, L. Plaza, and D. Camacho, "Combining graph connectivity and genetic clustering to improve biomedical summarization," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jul. 2014, pp. 2740–2747.
- [26] N. A. Ramu, M. S. Bandurupalli, M. S. S. Nekkanti, and G. Ramesh, "Summarization of research publications using automatic extraction," in *Proc. Int. Conf. Intell. Data Commun. Technol. Internet Things*. Cham, Switzerland: Springer, 2019, pp. 1–10.
- [27] J.-M. Torres-Moreno, "Single-document summarization," in *Automatic Text Summarization*. Hoboken, NJ, USA: Wiley, 2014, pp. 53–108.
- [28] C. Ma, W. E. Zhang, M. Guo, H. Wang, and Q. Z. Sheng, "Multi-document summarization via deep learning techniques: A survey," 2020, *arXiv:2011.04843*.
- [29] N. Ramanujam and M. Kaliappan, "An automatic multidocument text summarization approach based on Naïve Bayesian classifier using timestamp strategy," *Sci. World J.*, vol. 2016, pp. 1–10, Jan. 2016.
- [30] A. Fabbri, I. Li, T. She, S. Li, and D. Radev, "Multi-news: A large-scale multi-document summarization dataset and abstractive hierarchical model," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, 2019, pp. 1074–1084.
- [31] M. Yasunaga, J. Kasai, R. Zhang, A. R. Fabbri, I. Li, D. Friedman, and D. R. Radev, "Scisummnet: A large annotated corpus and content-impact models for scientific paper summarization with citation networks," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, no. 1, 2019, pp. 7386–7393.
- [32] G. Carenini, R. T. Ng, and X. Zhou, "Summarizing email conversations with clue words," in *Proc. 16th Int. Conf. World Wide Web (WWW)*, 2007, pp. 91–100.
- [33] D. M. Zajic, B. J. Dorr, and J. Lin, "Single-document and multi-document summarization techniques for email threads using sentence compression," *Inf. Process. Manage.*, vol. 44, no. 4, pp. 1600–1610, Jul. 2008.
- [34] S. Gerani, Y. Mehdad, G. Carenini, R. T. Ng, and B. Nejat, "Abstractive summarization of product reviews using discourse structure," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2014, pp. 1602–1613.
- [35] W. Luo, F. Liu, Z. Liu, and D. Litman, "Automatic summarization of student course feedback," 2018, *arXiv:1805.10395*.
- [36] W. Luo and D. Litman, "Summarizing student responses to reflection prompts," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2015, pp. 1955–1960.
- [37] P. J. Liu, M. Saleh, E. Pot, B. Goodrich, R. Sepassi, L. Kaiser, and N. Shazeer, "Generating Wikipedia by summarizing long sequences," 2018, *arXiv:1801.10198*.
- [38] S. Afantenos, V. Karkaletsis, and P. Stamatopoulos, "Summarization from medical documents: A survey," *Artif. Intell. Med.*, vol. 33, no. 2, pp. 157–177, Feb. 2005, doi: 10.1016/j.artmed.2004.07.017.
- [39] M. Alghamdi, C. Treude, and M. Wagner, "Human-like summaries from heterogeneous and time-windowed software development artefacts," Springer, USA, Tech. Rep., 2020.
- [40] Wikipedia Contributors. (2020). *Multi-Document Summarization—Wikipedia, the Free Encyclopedia*. Accessed: Oct. 8, 2021. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Multi-document_summarization%&oldid=986613170
- [41] H. Chen, "Digital library development in the Asia Pacific," in *Proc. Int. Conf. Asian Digital Libraries*. USA: Springer, 2005, pp. 509–524.
- [42] J.-Y. Yeh, H.-R. Ke, W.-P. Yang, and I.-H. Meng, "Text summarization using a trainable summarizer and latent semantic analysis," *Inf. Process. Manage.*, vol. 41, no. 1, pp. 75–95, Jan. 2005.
- [43] A. Moreno and T. Redondo, "Text analytics: The convergence of big data and artificial intelligence," *Int. J. Interact. Multimed. Artif. Intell.*, vol. 3, no. 6, pp. 57–64, 2016.
- [44] F.-R. Lin and C.-H. Liang, "Storyline-based summarization for news topic retrospection," *Decis. Support Syst.*, vol. 45, no. 3, pp. 473–490, 2008.
- [45] Y. Du, Q. Li, L. Wang, and Y. He, "Biomedical-domain pre-trained language model for extractive summarization," *Knowl.-Based Syst.*, vol. 199, Jul. 2020, Art. no. 105964.
- [46] W. Kryściński, B. McCann, C. Xiong, and R. Socher, "Evaluating the factual consistency of abstractive text summarization," 2019, *arXiv:1910.12840*.
- [47] Q. Zhou, N. Yang, F. Wei, and M. Zhou, "Selective encoding for abstractive sentence summarization," 2017, *arXiv:1704.07073*.
- [48] J. Lin, X. Sun, S. Ma, and Q. Su, "Global encoding for abstractive summarization," 2018, *arXiv:1805.03989*.
- [49] X. Shen, Y. Zhao, H. Su, and D. Klakow, "Improving latent alignment in text summarization by generalizing the pointer generator," in *Proc. Conf. Empirical Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process. (EMNLP-IJCNLP)*, 2019, pp. 3753–3764.
- [50] R. Nallapati, B. Zhou, C. Nogueira dos santos, C. Gulcehre, and B. Xiang, "Abstractive text summarization using sequence-to-sequence RNNs and beyond," 2016, *arXiv:1602.06023*.
- [51] D. Harman and P. Over, "The effects of human variation in DUC summarization evaluation," in *Text Summarization Branches Out*. Bangkok, Thailand: Assoc. Comput. Linguistics, 2004, pp. 10–17.
- [52] A. Abdi, S. M. Shamsuddin, S. Hasan, and J. Piran, "Automatic sentiment-oriented summarization of multi-documents using soft computing," *Soft Comput.*, vol. 23, no. 20, pp. 10551–10568, Oct. 2019.
- [53] C.-Y. Lin and E. Hovy, "From single to multi-document summarization," in *Proc. 40th Annu. Meeting Assoc. Comput. Linguistics (ACL)*, 2001, pp. 457–464.
- [54] I. V. Mashechkin, M. I. Petrovskiy, D. S. Popov, and D. V. Tsarev, "Automatic text summarization using latent semantic analysis," *Program. Comput. Softw.*, vol. 37, no. 6, pp. 299–305, Nov. 2011.

- [55] A. Nenkova, "Automatic text summarization of newswire: Lessons learned from the document understanding conference," Columbia Univ., New York, NY, USA, Tech. Rep., 2005.
- [56] C.-Y. Lin and E. Hovy, "Automated multi-document summarization in NeATS," in *Proc. 2nd Int. Conf. Human Lang. Technol. Res.*, 2002, pp. 23–27.
- [57] R. Ferreira, L. de Souza Cabral, F. Freitas, R. D. Lins, G. de França Silva, S. J. Simske, and L. Favaro, "A multi-document summarization system based on statistics and linguistic treatment," *Expert Syst. Appl.*, vol. 41, no. 13, pp. 5780–5787, 2014.
- [58] E. Lloret, O. Ferrández, R. Munoz, and M. Palomar, "A text summarization approach under the influence of textual entailment," in *Proc. NLPSCS*, 2008, pp. 22–31.
- [59] A. Sinha, A. Yadav, and A. Gahlot, "Extractive text summarization using neural networks," 2018, *arXiv:1802.10137*.
- [60] A. Patel, T. Siddiqui, and U. Tiwary, "A language independent approach to multilingual text summarization," in *Large Scale Semantic Access to Content (Text, Image, Video, Sound)*, 2007, pp. 123–132.
- [61] H. Asgari, B. Masoumi, and O. S. Sheijani, "Automatic text summarization based on multi-agent particle swarm optimization," in *Proc. Iranian Conf. Intell. Syst. (ICIS)*, Feb. 2014, pp. 1–5.
- [62] M. S. Binwahlen, N. Salim, and L. Suamali, "Swarm based features selection for text summarization," *IJCSNS Int. J. Comput. Sci. Netw. Secur.*, vol. 9, no. 1, pp. 175–179, 2009.
- [63] P.-Y. Zhang and C.-H. Li, "Automatic text summarization based on sentences clustering and extraction," in *Proc. 2nd IEEE Int. Conf. Comput. Sci. Inf. Technol.*, Aug. 2009, pp. 167–170.
- [64] D. R. Radev, T. Allison, S. Blair-Goldensohn, J. Blitzer, A. Celebi, S. Dimitrov, E. Drabek, A. Hakim, W. Lam, D. Liu, and J. Otterbacher, "MEAD—A platform for multidocument multilingual text summarization," ACLI Anthology, Lisbon, Portugal, Tech. Rep., 2004.
- [65] G. Erkan and D. R. Radev, "LexRank: Graph-based lexical centrality as salience in text summarization," *J. Artif. Intell. Res.*, vol. 22, pp. 457–479, Dec. 2004.
- [66] E. H. Hovy, C.-Y. Lin, L. Zhou, and J. Fukumoto, "Automated summarization evaluation with basic elements," in *Proc. LREC*, vol. 6, Princeton, NJ, USA: Citeseer, 2006, pp. 604–611.
- [67] H. Takamura and M. Okumura, "Text summarization model based on maximum coverage problem and its variant," in *Proc. 12th Conf. Eur. Chapter Assoc. Comput. Linguistics (EACL)*, 2009, pp. 781–789.
- [68] G. Erkan and D. Radev, "Lexpagerank: Prestige in multi-document text summarization," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2004, pp. 365–371.
- [69] A. Nenkova and L. Vanderwende, "The impact of frequency on summarization," Microsoft Res., Redmond, WA, USA, Tech. Rep. MSR-TR-2005, 2005, vol. 101.
- [70] X. Wan and J. Yang, "Improved affinity graph based multi-document summarization," in *Proc. Hum. Lang. Technol. Conf. NAACL, Companion, Short Papers (XX-NAACL)*, 2006, pp. 181–184.
- [71] A. Dasgupta, R. Kumar, and S. Ravi, "Summarization through submodularity and dispersion," in *Proc. 51st Annu. Meeting Assoc. Comput. Linguistics, (Long Papers)*, vol. 1, 2013, pp. 1014–1022.
- [72] J. M. Conroy, J. D. Schlesinger, J. Goldstein, and D. P. O'leary, "Left-brain/right-brain multi-document summarization," in *Proc. Document Understand. Conf. (DUC)*, May 2004, pp. 1–9.
- [73] Y. Chali and M. Kolla, "Summarization techniques at DUC 2004," in *Proc. Document Understand. Conf.*, Gaithersburg, MD, USA: National Institute of Standards in Technology (NIST), 2004, pp. 105–111.
- [74] K. C. Litkowski, "Summarization experiments in DUC 2004," in *Proc. Document Understand. Conf.*, 2004, pp. 6–7.
- [75] H. T. Dang, "Overview of DUC 2005," in *Proc. Document Understand. Conf.*, 2005, pp. 1–12.
- [76] G. Melli, Y. Wang, Y. Liu, M. M. Kashani, Z. Shi, B. Gu, A. Sarkar, and F. Popowich, "Description of squash, the SFU question answering summary handler for the DUC-2005 summarization task," *Safety*, vol. 1, Jan. 2005, Art. no. 14345754.
- [77] S. Li, Y. Ouyang, W. Wang, and B. Sun, "Multi-document summarization using support vector regression," in *Proc. DUC*, Princeton, NJ, USA: Citeseer, 2007, pp. 1–5.
- [78] E. Hovy, C.-Y. Lin, and L. Zhou, "Evaluating DUC 2005 using basic elements," in *Proc. DUC*, Princeton, NJ, USA: Citeseer, 2005, pp. 1–6.
- [79] A. Mohamed and S. Rajasekaran, "Improving query-based summarization using document graphs," in *Proc. IEEE Int. Symp. Signal Process. Inf. Technol.*, Aug. 2006, pp. 408–410.
- [80] Y. Seki, K. Eguchi, N. Kando, and M. Aono, "Opinion-focused summarization and its analysis at DUC 2006," in *Proc. Document Understand. Conf. (Duc)*, 2006, pp. 122–130.
- [81] A. Haghighi and L. Vanderwende, "Exploring content models for multi-document summarization," in *Proc. Human Lang. Technol., Annu. Conf. North Amer. Chapter Assoc. Comput. Linguistics (NAACL)*, 2009, pp. 362–370.
- [82] F. Lacatusu, A. Hickl, K. Roberts, Y. Shi, J. Bensley, B. Rink, P. Wang, and L. Taylor, "LCC's GISTexter at DUC 2006: Multi-strategy multi-document summarization," in *Proc. DUC*, 2006, pp. 1–9.
- [83] M. R. Amini and N. Usunier, "A contextual query expansion approach by term clustering for robust text summarization," in *Proc. DUC*, vol. 200, no. 7, Princeton, NJ, USA: Citeseer, 2007, pp. 1–8.
- [84] P. Over, H. Dang, and D. Harman, "DUC in context," *Inf. Process. Manage.*, vol. 43, no. 6, pp. 1506–1520, Nov. 2007.
- [85] M. A. Tayal, M. M. Raghuwanshi, and L. G. Malik, "ATSSC: Development of an approach based on soft computing for text summarization," *Comput. Speech Lang.*, vol. 41, pp. 214–235, Jan. 2017.
- [86] R. Verma, P. Chen, and W. Lu, "A semantic free-text summarization system using ontology knowledge," in *Proc. Document Understand. Conf.*, 2007, pp. 1–5.
- [87] K. Toutanova, C. Brockett, M. Gamon, J. Jagarlamudi, H. Suzuki, and L. Vanderwende, "The PYTHY summarization system: Microsoft research at DUC 2007," in *Proc. DUC*, 2007, pp. 1–8.
- [88] M. Afsharizadeh, H. Ebrahimpour-Komleh, and A. Bagheri, "Query-oriented text summarization using sentence extraction technique," in *Proc. 4th Int. Conf. Web Res. (ICWR)*, Apr. 2018, pp. 128–132.
- [89] N. Madnani, D. Zajic, B. Dorr, N. F. Ayan, and J. Lin, "Multiple alternative sentence compressions for automatic text summarization," in *Proc. Document Understand. Conf. (DUC) (NLT/NAACL)*, 2007, p. 24.
- [90] F. T. AL-Khawaldeh and V. W. Samawi, "Lexical cohesion and entailment based segmentation for Arabic text summarization (LCEAS)," *World Comput. Sci. Inf. Technol. J.*, vol. 5, no. 3, pp. 1–10, 2015.
- [91] R. Z. Al-Abdallah and A. T. Al-Taani, "Arabic single-document text summarization using particle swarm optimization algorithm," *Proc. Comput. Sci.*, vol. 117, pp. 30–37, Jan. 2017.
- [92] H. Oufaida, O. Nouali, and P. Blache, "Minimum redundancy and maximum relevance for single and multi-document Arabic text summarization," *J. King Saud Univ. Comput. Inf. Sci.*, vol. 26, no. 4, pp. 450–461, Dec. 2014.
- [93] L. D. S. Cabral, R. D. Lins, R. F. Mello, F. Freitas, B. Ávila, S. Simske, and M. Riss, "A platform for language independent summarization," in *Proc. ACM Symp. Document Eng. (DocEng)*, 2014, pp. 203–206.
- [94] R. Mihalcea, "Language independent extractive summarization," in *Proc. ACL Interact. Poster Demonstration Sessions (ACL)*, 2005, pp. 49–52.
- [95] D. R. Amancio, M. G. V. Nunes, O. N. Oliveira, and L. D. F. Costa, "Extractive summarization using complex networks and syntactic dependency," *Phys. A, Stat. Mech. Appl.*, vol. 391, no. 4, pp. 1855–1864, Feb. 2012.
- [96] T. S. Pardo, L. Antiquera, M. G. V. Nunes, O. Oliveira, and L. F. Costa, "Using complex networks for language processing: The case of summary evaluation," in *Proc. Int. Conf. Commun., Circuits Syst.*, Jun. 2006, pp. 2678–2682.
- [97] L. H. M. Rino, T. A. S. Pardo, C. N. Silla, C. A. A. Kaestner, and M. Pombo, "A comparison of automatic summarizers of texts in Brazilian Portuguese," in *Proc. Brazilian Symp. Artif. Intell.*, Berlin, Germany: Springer, 2004, pp. 235–244.
- [98] S. Liu, M. X. Zhou, S. Pan, Y. Song, W. Qian, W. Cai, and X. Lian, "TIARA: Interactive, topic-based visual text summarization and analysis," *ACM Trans. Intell. Syst. Technol. (TIST)*, vol. 3, no. 2, pp. 1–28, Feb. 2012.
- [99] M. Yousefi-Azar and L. Hamey, "Text summarization using unsupervised deep learning," *Expert Syst. Appl.*, vol. 68, pp. 93–105, Feb. 2017.
- [100] J. Ulrich, G. Murray, and G. Carenini, "A publicly available annotated corpus for supervised email summarization," in *Proc. AAAI Email Workshop*, 2008, pp. 1–6.
- [101] N. Alami, M. Meknassi, and N. En-Nahnah, "Enhancing unsupervised neural networks based text summarization with word embedding and ensemble learning," *Expert Syst. Appl.*, vol. 123, pp. 195–211, Jun. 2019.
- [102] M. Yang, X. Wang, Y. Lu, J. Lv, Y. Shen, and C. Li, "Plausibility-promoting generative adversarial network for abstractive text summarization with multi-task constraint," *Inf. Sci.*, vol. 521, pp. 46–61, Jun. 2020.

- [103] H. Zhang, J. Xu, and J. Wang, "Pretraining-based natural language generation for text summarization," 2019, *arXiv:1902.09243*.
- [104] U. Khandelwal, K. Clark, D. Jurafsky, and L. Kaiser, "Sample efficient text summarization using a single pre-trained transformer," 2019, *arXiv:1905.08836*.
- [105] C. Li, W. Xu, S. Li, and S. Gao, "Guiding generation for abstractive text summarization based on key information guide network," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Human Lang. Technol., (Short Papers)*, vol. 2, 2018, pp. 55–60.
- [106] R. Paulus, C. Xiong, and R. Socher, "A deep reinforced model for abstractive summarization," 2017, *arXiv:1705.04304*.
- [107] Q. Zhou, N. Yang, F. Wei, S. Huang, M. Zhou, and T. Zhao, "Neural document summarization by jointly learning to score and select sentences," 2018, *arXiv:1807.02305*.
- [108] M. Koupaei and W. Y. Wang, "WikiHow: A large scale text summarization dataset," 2018, *arXiv:1810.09305*.
- [109] J. Zhang, Y. Zhao, M. Saleh, and P. Liu, "Pegasus: Pre-training with extracted gap-sentences for abstractive summarization," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 11328–11339.
- [110] I. Mani, G. Klein, D. House, L. Hirschman, T. Firmin, and B. Sundheim, "SUMMAC: A text summarization evaluation," *Natural Lang. Eng.*, vol. 8, no. 1, pp. 43–68, Mar. 2002.
- [111] I. Mani, D. House, G. Klein, L. Hirschman, T. Firmin, and B. Sundheim, "The TIPSTER SUMMAC text summarization evaluation," in *Proc. 9th Conf. Eur. Chapter Assoc. Comput. Linguistics*, 1999, pp. 77–85.
- [112] S. Azzam, K. Humphreys, and R. Gaizauskas, "Using coreference chains for text summarization," in *Proc. Workshop Coreference Appl. (CorefApp)*, 1999, pp. 1–8.
- [113] M. Okumura, T. Fukusima, and H. Nanba, "Text summarization challenge 2: Text summarization evaluation at NTCIR workshop 3," in *Proc. HLT-NAACL Text Summarization Workshop*, 2003, pp. 51–59.
- [114] E. Hovy and D. Marcu, "Automated text summarization," in *The Oxford Handbook of Computational Linguistics*, vol. 583598. Oxford, U.K.: Oxford Univ. Press, 2005.
- [115] T. Nomoto and Y. Matsumoto, "A new approach to unsupervised text summarization," in *Proc. 24th Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retr. (SIGIR)*, 2001, pp. 26–34.
- [116] O. Tas and F. Kiyani, "A survey automatic text summarization," *Pressacademia*, vol. 5, no. 1, pp. 205–213, Jun. 2017.
- [117] H. Saggon and T. Poibeau, "Automatic text summarization: Past, present and future," in *Multi-Source, Multilingual Information Extraction and Summarization*. Berlin, Germany: Springer, 2013, pp. 3–21.
- [118] H. Zhang and J. Wang, "Semantic wordrank: Generating finer single-document summarizations," in *Proc. Int. Conf. Intell. Data Eng. Automated Learn.* Cham, Switzerland: Springer, 2018, pp. 398–409.
- [119] L. Shao, H. Zhang, and J. Wang, "Robust single-document summarizations and a semantic measurement of quality," in *Proc. Int. Joint Conf. Knowl. Discovery, Knowl. Eng., Knowl. Manage.* Cham, Switzerland: Springer, 2017, pp. 118–138.
- [120] K. Ganesan, C. Zhai, and J. Han, "Opinosis: A graph based approach to abstractive summarization of highly redundant opinions," Illinois Digit. Environ. Access Learn. Scholarship, USA, Tech. Rep. 148, 2010.
- [121] M. Kågeback, O. Mogren, N. Tahmasebi, and D. Dubhashi, "Extractive summarization using continuous vector space models," in *Proc. 2nd Workshop Continuous Vector Space Models Their Compositionality (CVSC)*, 2014, pp. 31–39.
- [122] A. Padmakumar and A. Saran, "Unsupervised text summarization using sentence embeddings," Dept. Comput. Sci., Univ. Texas, Austin, TX, USA, Tech. Rep., 2016, [Online]. Available: <https://www.cs.utexas.edu/~aish/ut/NLPPProject.pdf>
- [123] D. Gillick, B. Favre, and D. Hakkani-Tür, "The ICSI summarization system at TAC 2008," in *Proc. TAC*, 2008, pp. 1–8.
- [124] D. Galanis and P. Malakasiotis, "AUEB at TAC 2008," in *Proc. TAC*, 2008, pp. 1–7.
- [125] A. Bawakid and M. Oussalah, "A semantic summarization system: University of Birmingham at TAC 2008," in *Proc. TAC*. Princeton, NJ, USA: Citeseer, 2008, pp. 1–6.
- [126] M. Bhandari, P. Gour, A. Ashfaq, P. Liu, and G. Neubig, "Re-evaluating evaluation in text summarization," 2020, *arXiv:2010.07100*.
- [127] D. Yogatama, F. Liu, and N. A. Smith, "Extractive summarization by maximizing semantic volume," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2015, pp. 1961–1966.
- [128] M. Peyrard, "Studying summarization evaluation metrics in the appropriate scoring range," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, 2019, pp. 5093–5100.
- [129] L. Marujo, W. Ling, R. Ribeiro, A. Gershman, J. Carbonell, D. M. De Matos, and J. P. Neto, "Exploring events and distributed representations of text in multi-document summarization," *Knowl.-Based Syst.*, vol. 94, pp. 33–42, Feb. 2016.
- [130] V. Varma, P. Pingali, R. Katragadda, S. Krishna, S. Ganesh, K. Sarvabhotla, H. Garapati, H. Gopisetty, V. B. Reddy, K. Reddy, and P. Bysani, "IIIT Hyderabad at TAC 2009," in *Proc. TAC*, Nov. 2009, pp. 1–6.
- [131] J.-P. Ng and V. Abrecht, "Better summarization evaluation with word embeddings for Rouge," 2015, *arXiv:1508.06034*.
- [132] D. Gillick, B. Favre, D. Hakkani-Tür, B. Bohnet, Y. Liu, and S. Xie, "The ICSI/UTD summarization system at TAC 2009," in *Proc. TAC*. Princeton, NJ, USA: Citeseer, 2009, pp. 1–8.
- [133] K. Hong, M. Marcus, and A. Nenkova, "System combination for multi-document summarization," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2015, pp. 107–117.
- [134] H. Ji, R. Grishman, H. T. Dang, K. Griffith, and J. Ellis, "Overview of the TAC 2010 knowledge base population track," in *Proc. 3rd text Anal. Conf. (TAC)*, vol. 3, no. 2, 2010, p. 3.
- [135] J.-Y. Delort and E. Alfonseca, "Dualsum: A topic-model based approach for update summarization," in *Proc. 13th Conf. Eur. Chapter Assoc. Comput. Linguistics*, 2012, pp. 214–223.
- [136] F. Jin, M. Huang, and X. Zhu, "The THU summarization systems at TAC 2010," in *Proc. TAC*. Princeton, NJ, USA: Citeseer, 2010, pp. 1–6.
- [137] A. Kennedy, A. Kazantseva, D. Inkpen, and S. Szpakowicz, "Getting emotional about news summarization," in *Proc. Can. Conf. Artif. Intell.* Berlin, Germany: Springer, 2012, pp. 121–132.
- [138] W. M. Darling and F. Song, "PathSum: A summarization framework based on hierarchical topics," Autom. Text Summarization, Dept. Comput. Sci., Univ. Toronto, Toronto, ON, Canada, 2011, p. 5.
- [139] T. Makino, H. Takamura, and M. Okumura, "Balanced coverage of aspects for text summarization," in *Proc. 21st ACM Int. Conf. Inf. Knowl. Manage. (CIKM)*, 2012, pp. 1742–1746.
- [140] G. Giannakopoulos, M. El-Haj, B. Favre, M. Litvak, J. Steinberger, and V. Varma, "TAC 2011 multiling pilot overview," in *Proc. Text Anal. Conf. (TAC)*, MD, USA, 2011.
- [141] G. Algaphari, F. M. Ba-Alwi, and A. Moharram, "Text summarization using centrality concept," *Int. J. Comput. Appl.*, vol. 79, no. 1, pp. 5–12, Oct. 2013.
- [142] K. Owczarzak, J. Conroy, H. T. Dang, and A. Nenkova, "An assessment of the accuracy of automatic evaluation in summarization," in *Proc. Workshop Eval. Metrics Syst. Comparison Autom. Summarization*, 2012, pp. 1–9.
- [143] A. Dean-Hall, C. L. Clarke, J. Kamps, P. Thomas, N. Simone, and E. Voorhees, "Overview of the TREC 2013 contextual suggestion track," Waterloo Univ., Waterloo, ON, Canada, Tech. Rep. ADA599415, 2013.
- [144] F. Démoncourt, M. Ghassemi, and W. Chang, "A repository of corpora for summarization," in *Proc. 11th Int. Conf. Lang. Resour. Eval. (LREC)*, 2018, pp. 1–7.
- [145] R. McCreadie, C. Macdonald, and I. Ounis, "Incremental update summarization: Adaptive sentence selection based on prevalence and novelty," in *Proc. 23rd ACM Int. Conf. Conf. Inf. Knowl. Manage.*, Nov. 2014, pp. 301–310.
- [146] A. Kanapala, S. Pal, and R. Pamula, "Text summarization from legal documents: A survey," *Artif. Intell. Rev.*, vol. 51, no. 3, pp. 371–402, Mar. 2019.
- [147] N. Moratanch and S. Chitrakala, "A survey on abstractive text summarization," in *Proc. Int. Conf. Circuit, Power Comput. Technol. (ICCPCT)*, Mar. 2016, pp. 1–7.
- [148] Y. Zhao, F. Yao, H. Sun, and Z. Yang, "BJUT at TREC 2014 temporal summarization track," Beijing Univ. Technol., Beijing, China, Tech. Rep. ADA618573, 2014.
- [149] H. Li, Z. Yang, Y. Lai, L. Duan, and K. Fan, "BJUT at TREC 2014 contextual suggestion track: Hybrid recommendation based on open-web information," Beijing Univ. Technol., Beijing, China, Tech. Rep. ADA618571, 2014.
- [150] M. Aliannejadi, S. A. Bahrainian, A. Giachanou, and F. Crestani, "University of Lugano at TREC 2015: Contextual suggestion and temporal summarization tracks," in *Proc. TREC*, 2015, pp. 1–6.
- [151] E. Agichtein, D. Carmel, D. Pelleg, Y. Pinter, and D. Harman, "Overview of the TREC 2015 LiveQA track," in *Proc. TREC*, 2015, pp. 1–9.
- [152] N. Moratanch and S. Chitrakala, "A survey on extractive text summarization," in *Proc. Int. Conf. Comput., Commun. Signal Process. (ICCCSP)*, Jan. 2017, pp. 1–6.
- [153] B. Hu, Q. Chen, and F. Zhu, "LCSTS: A large scale Chinese short text summarization dataset," 2015, *arXiv:1506.05865*.

- [154] L. Wang, J. Yao, Y. Tao, L. Zhong, W. Liu, and Q. Du, "A reinforced topic-aware convolutional sequence-to-sequence model for abstractive text summarization," 2018, *arXiv:1805.03616*.
- [155] P. Li, W. Lam, L. Bing, and Z. Wang, "Deep recurrent generative decoder for abstractive text summarization," 2017, *arXiv:1708.00625*.
- [156] Z. Li, Z. Peng, S. Tang, C. Zhang, and H. Ma, "Text summarization method based on double attention pointer network," *IEEE Access*, vol. 8, pp. 11279–11288, 2020.
- [157] Z. Liang, J. Du, and C. Li, "Abstractive social media text summarization using selective reinforced Seq2Seq attention model," *Neurocomputing*, vol. 410, pp. 432–440, Oct. 2020.
- [158] S. Ma and X. Sun, "A semantic relevance based neural network for text summarization and text simplification," 2017, *arXiv:1710.02318*.
- [159] A. Joshi, E. Fidalgo, E. Alegria, and L. Fernández-Robles, "SumCoder: An unsupervised framework for extractive text summarization based on deep auto-encoders," *Expert Syst. Appl.*, vol. 129, pp. 200–215, Sep. 2019.
- [160] L.-W. Ku, Y. T. Liang, and H. H. Chen, "Opinion extraction, summarization and tracking in news and blog corpora," in *Proc. AAAI Spring Symp., Comput. Approaches Analyzing Weblogs*, vol. 100107, 2006, pp. 1–167.
- [161] M. Hu, A. Sun, and E.-P. Lim, "Comments-oriented blog summarization by sentence extraction," in *Proc. 16th ACM Conf. Conf. Inf. Knowl. Manage. (CIKM)*, 2007, pp. 901–904.
- [162] G. Kim, S. Moon, and L. Sigal, "Joint photo stream and blog post summarization and exploration," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3081–3089.
- [163] S. Petrov, D. Das, and R. McDonald, "A universal part-of-speech tagset," 2011, *arXiv:1104.2086*.
- [164] J. R. Méndez, E. L. Iglesias, F. Fdez-Riverola, F. Díaz, and J. M. Corchado, "Tokenising, stemming and stopword removal on anti-spam filtering domain," in *Proc. Conf. Spanish Assoc. Artif. Intell.* Berlin, Germany: Springer, 2005, pp. 449–458.
- [165] A. Mansouri, L. S. Affendey, and A. Mamat, "Named entity recognition approaches," *Int. J. Comput. Sci. Netw. Secur.*, vol. 8, no. 2, pp. 339–344, 2008.
- [166] G. Gupta and S. Malhotra, "Text document tokenization for word frequency count using rapid miner (taking resume as an example)," *Int. J. Comput. Appl.*, vol. 975, p. 8887, Jan. 2015.
- [167] T. Verma, R. Renu, and D. Gaur, "Tokenization and filtering process in RapidMiner," *Int. J. Appl. Inf. Syst.*, vol. 7, no. 2, pp. 16–18, Apr. 2014.
- [168] K. Darwish, "Named entity recognition using cross-lingual resources: Arabic as an example," in *Proc. 51st Annu. Meeting Assoc. Comput. Linguistics (Long Papers)*, vol. 1, 2013, pp. 1558–1567.
- [169] D. L. Whitney and B. W. Evans, "Abbreviations for names of rock-forming minerals," *Amer. Mineralog.*, vol. 95, no. 1, pp. 185–187, Jan. 2010.
- [170] B. Pahwa, S. Taruna, and N. Kasliwal, "Sentiment analysis—Strategy for text pre-processing," *Int. J. Comput. Appl.*, vol. 180, no. 34, pp. 15–18, Apr. 2018.
- [171] J. Schaback and F. Li, "Multi-level feature extraction for spelling correction," in *Proc. IJCAI Workshop Anal. Noisy Unstructured Text Data*. Princeton, NJ, USA: Citeseer, 2007, pp. 79–86.
- [172] J. Plisson, N. Lavrac, and D. Mladenic, "A rule based approach to word lemmatization," in *Proc. IS*, vol. 3, 2004, pp. 83–86.
- [173] T. Korenius, J. Laurikkala, K. Järvelin, and M. Juhola, "Stemming and lemmatization in the clustering of Finnish text documents," in *Proc. 13th ACM Conf. Inf. Knowl. Manage. (CIKM)*, 2004, pp. 625–633.
- [174] J. L. Neto, A. A. Freitas, and C. A. Kaestner, "Automatic text summarization using a machine learning approach," in *Proc. Brazilian Symp. Artif. Intell.* Berlin, Germany: Springer, 2002, pp. 205–215.
- [175] H. Christian, M. P. Agus, and D. Suhartono, "Single document automatic text summarization using term frequency-inverse document frequency (TF-IDF)," *ComTech, Comput., Math. Eng. Appl.*, vol. 7, no. 4, pp. 285–294, 2016.
- [176] M. A. Fattah and F. Ren, "GA, MR, FFNN, PNN and GMM based models for automatic text summarization," *Comput. Speech Lang.*, vol. 23, no. 1, pp. 126–144, Jan. 2009.
- [177] M. A. Fattah and F. Ren, "Automatic text summarization," *World Academy Sci., Eng. Technol.*, vol. 37, no. 2, p. 192, 2008.
- [178] B. Mutlu, E. A. Sezer, and M. A. Akcayol, "Multi-document extractive text summarization: A comparative assessment on features," *Knowl.-Based Syst.*, vol. 183, Nov. 2019, Art. no. 104848.
- [179] S. A. Babar and P. D. Patil, "Improving performance of text summarization," *Proc. Comput. Sci.*, vol. 46, pp. 354–363, Jan. 2015.
- [180] Y. Mehdad, G. Carenini, and R. T. Ng, "Abstractive summarization of spoken and written conversations based on phrasal queries," in *Proc. 52nd Annu. Meeting Assoc. Comput. Linguistics (Long Papers)*, vol. 1, 2014, pp. 1220–1230.
- [181] M. Abdolahi and M. Zahedh, "Sentence matrix normalization using most likely n-grams vector," in *Proc. IEEE 4th Int. Conf. Knowl.-Based Eng. Innov. (KBEI)*, Dec. 2017, pp. 40–45.
- [182] C.-Y. Lin and E. Hovy, "The automated acquisition of topic signatures for text summarization," in *Proc. 18th Conf. Comput. linguistics*, 2000, pp. 1–7.
- [183] L. Suanmali, M. S. Binwahlan, and N. Salim, "Sentence features fusion for text summarization using fuzzy logic," in *Proc. 9th Int. Conf. Hybrid Intell. Syst.*, 2009, pp. 142–146.
- [184] Y. Ma and J. Wu, "Combining N-Gram and dependency word pair for multi-document summarization," in *Proc. IEEE 17th Int. Conf. Comput. Sci. Eng.*, Dec. 2014, pp. 27–31.
- [185] G. Giannakopoulos, V. Karkaletsis, G. Vouros, and P. Stamatopoulos, "Summarization system evaluation revisited: N-gram graphs," *ACM Trans. Speech Lang. Process.*, vol. 5, no. 3, pp. 1–39, Oct. 2008.
- [186] L. Hennig, W. Umbrath, and R. Wetzker, "An ontology-based approach to text summarization," in *Proc. IEEE/WIC/ACM Int. Conf. Web Intell. Intell. Agent Technol.*, Dec. 2008, pp. 291–294.
- [187] K. Mani, I. Verma, H. Meisheri, and L. Dey, "Multi-document summarization using distributed bag-of-words model," in *Proc. IEEE/WIC/ACM Int. Conf. Web Intell. (WI)*, Dec. 2018, pp. 672–675.
- [188] D. Yan, K. Li, S. Gu, and L. Yang, "Network-based bag-of-words model for text classification," *IEEE Access*, vol. 8, pp. 82641–82652, 2020.
- [189] K. Spärck Jones, "A statistical interpretation of term specificity and its application in retrieval," *J. Document.*, vol. 60, no. 5, pp. 493–502, Oct. 2004.
- [190] N. Alsaedi, P. Burnap, and O. Rana, "Temporal TF-IDF: A high performance approach for event summarization in Twitter," in *Proc. IEEE/WIC/ACM Int. Conf. Web Intell. (WI)*, Oct. 2016, pp. 515–521.
- [191] J. Steinberger and K. Jezek, "Using latent semantic analysis in text summarization and summary evaluation," in *Proc. ISIM*, vol. 4, 2004, pp. 93–100.
- [192] J. L. Neto, A. D. Santos, C. A. Kaestner, N. Alexandre, and D. Santos, "Document clustering and text summarization," 2000.
- [193] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 3111–3119.
- [194] Y. Wang, X. Liu, and Z. Gao, "Neural related work summarization with a joint context-driven attention mechanism," 2019, *arXiv:1901.09492*.
- [195] M. M. Haider, M. A. Hossin, H. R. Mahi, and H. Arif, "Automatic text summarization using Gensim Word2 Vec and K-means clustering algorithm," in *Proc. IEEE Region 10 Symp. (TENSYP)*, Jun. 2020, pp. 283–286.
- [196] S. Abdulateef, N. A. Khan, B. Chen, and X. Shang, "Multidocument Arabic text summarization based on clustering and Word2 Vec to reduce redundancy," *Information*, vol. 11, no. 2, p. 59, Jan. 2020.
- [197] J. Niu, M. Sun, J. J. P. C. Rodrigues, and X. Liu, "A novel attention mechanism considering decoder input for abstractive text summarization," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2019, pp. 1–7.
- [198] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2014, pp. 1532–1543.
- [199] A. Jain, D. Bhatia, and M. K. Thakur, "Extractive text summarization using word vector embedding," in *Proc. Int. Conf. Mach. Learn. Data Sci. (MLDS)*, Dec. 2017, pp. 51–55.
- [200] S. Villata, "Using argument mining for legal text summarization," in *Proc. Legal Knowl. Inf. Syst., JURIX, 33rd Annu. Conf.*, vol. 334. Brno, Czech Republic: IOS Press, Dec. 2020, p. 184.
- [201] C. Zhang, S. Sah, T. Nguyen, D. Peri, A. Loui, C. Salvaggio, and R. Ptucha, "Semantic sentence embeddings for paraphrasing and text summarization," in *Proc. IEEE Global Conf. Signal Inf. Process. (GlobalSIP)*, Nov. 2017, pp. 705–709.
- [202] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," *Trans. Assoc. Comput. Linguistics*, vol. 5, pp. 135–146, Dec. 2017.
- [203] K. Kurniawan and S. Louvan, "Indosum: A new benchmark dataset for Indonesian text summarization," in *Proc. Int. Conf. Asian Lang. Process. (IALP)*, Nov. 2018, pp. 215–220.

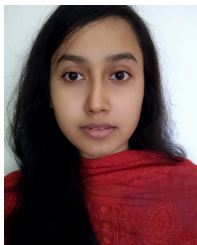
- 156067

- [255] P.-E. Genest and G. Lapalme, "Framework for abstractive summarization using text-to-text generation," in *Proc. Workshop Monolingual Text Text Gener.*, 2011, pp. 64–73.
- [256] P. E. Genest and G. Lapalme, "Fully abstractive approach to guided summarization," in *Proc. 50th Annu. Meeting Assoc. Comput. Linguistics*, vol. 2, Stroudsburg, PA, USA: Association Computational Linguistics, 2012, pp. 354–358.
- [257] J. Otterbacher, G. Erkan, and D. R. Radev, "Biased LexRank: Passage retrieval using random walks with question-based priors," *Inf. Process. Manage.*, vol. 45, no. 1, pp. 42–54, Jan. 2009.
- [258] K. Wu, P. Shi, and D. Pan, "An approach to automatic summarization for Chinese text based on the combination of spectral clustering and LexRank," in *Proc. 12th Int. Conf. Fuzzy Syst. Knowl. Discovery (FSKD)*, Aug. 2015, pp. 1350–1354.
- [259] P. Verma and H. Om, "Extraction based text summarization methods on user's review data: A comparative study," in *Proc. Int. Conf. Smart Trends Inf. Technol. Comput. Commun.* Singapore: Springer, 2016, pp. 346–354.
- [260] A. Li, T. Jiang, Q. Wang, and H. Yu, "The mixture of textrank and lexrank techniques of single document automatic summarization research in Tibetan," in *Proc. 8th Int. Conf. Intell. Hum.-Mach. Syst. Cybern. (IHMSC)*, Aug. 2016, pp. 514–519.
- [261] A. Samuel and D. K. Sharma, "Modified LexRank for tweet summarization," *Int. J. Rough Sets Data Anal.*, vol. 3, no. 4, pp. 79–90, Oct. 2016.
- [262] M. Litvak and M. Last, "Graph-based keyword extraction for single-document summarization," in *Proc. Workshop Multi-source Multilingual Inf. Extraction Summarization (MMIES)*, 2008, pp. 17–24.
- [263] K. S. Thakkar, R. V. Dharaskar, and M. B. Chandak, "Graph-based algorithms for text summarization," in *Proc. 3rd Int. Conf. Emerg. Trends Eng. Technol.*, Nov. 2010, pp. 516–519.
- [264] R. Mihalcea, "Graph-based ranking algorithms for sentence extraction, applied to text summarization," in *Proc. ACL Interact. Poster Demonstration Sessions*, 2004, pp. 170–173.
- [265] H. Bhandari, M. Shimbo, T. Ito, and Y. Matsumoto, "Generic text summarization using probabilistic latent semantic indexing," in *Proc. 3rd Int. Joint Conf. Natural Lang. Process.*, vol. 1, 2008, pp. 1–8.
- [266] J. Zhanq, L. Sun, and Q. Zhou, "A cue-based hub-authority approach for multi-document text summarization," in *Proc. Int. Conf. Natural Lang. Process. Knowl. Eng.*, Nov. 2005, pp. 642–645.
- [267] L. Page, S. Brin, R. Motwani, and T. Winograd, "The pagerank citation ranking: Bringing order to the web," Stanford InfoLab, Stanford, CA, USA, Tech. Rep., 1999.
- [268] R. Elbarougy, G. Behery, and A. El Khatib, "Extractive Arabic text summarization using modified PageRank algorithm," *Egyptian Informat. J.*, vol. 21, no. 2, pp. 73–81, Jul. 2020.
- [269] C. Mallick, A. K. Das, M. Dutta, A. K. Das, and A. Sarkar, "Graph-based text summarization using modified textrank," in *Soft Computing in Data Analytics*. Singapore: Springer, 2019, pp. 137–146.
- [270] J. Wang, J. Liu, and C. Wang, "Keyword extraction based on pagerank," in *Proc. Pacific-Asia Conf. Knowl. Discovery Data Mining*. Berlin, Germany: Springer, 2007, pp. 857–864.
- [271] A. Chongsuntornsri and O. Sornil, "An automatic Thai text summarization using topic sensitive PageRank," in *Proc. Int. Symp. Commun. Inf. Technol.*, Oct. 2006, pp. 547–552.
- [272] O. Sornil and K. Gree-Ut, "An automatic text summarization approach using content-based and graph-based characteristics," in *Proc. IEEE Conf. Cybern. Intell. Syst.*, Jun. 2006, pp. 1–6.
- [273] R. Mihalcea and P. Tarau, "TextRank: Bringing order into text," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2004, pp. 404–411.
- [274] F. Barrios, F. López, L. Argerich, and R. Wachenchauser, "Variations of the similarity function of TextRank for automated summarization," 2016, [arXiv:1602.03606](https://arxiv.org/abs/1602.03606).
- [275] N. Akhtar, M. M. S. Beg, and H. Javed, "TextRank enhanced topic model for query focussed text summarization," in *Proc. 12th Int. Conf. Contemp. Comput. (IC3)*, Aug. 2019, pp. 1–6.
- [276] D. Gunawan, A. Pasaribu, R. Rahmat, and R. Budiarto, "Automatic text summarization for Indonesian language using textteaser," in *Proc. IOP Conf. Ser., Mater. Sci. Eng.*, vol. 190, no. 1. Bristol, U.K.: IOP Publishing, 2017, Art. no. 012048.
- [277] N. Yadav and N. Chatterjee, "Text summarization using sentiment analysis for DUC data," in *Proc. Int. Conf. Inf. Technol. (ICIT)*, Dec. 2016, pp. 229–234.
- [278] N. K. Jha and A. Mitra, "Introducing word's importance level-based text summarization using tree structure," *Int. J. Inf. Retr. Res.*, vol. 10, no. 1, pp. 13–33, Jan. 2020.
- [279] K. Almekhlafi, "A review of graph-based extractive text summarization models," in *Innovative Systems for Intelligent Health Informatics: Data Science, Health Informatics, Intelligent Systems, Smart Computing*, vol. 72, no. 1. Cham, Switzerland: Springer, Dec. 2020, pp. 439–448.
- [280] L. Antiquiera, O. N. Oliveira, L. D. F. Costa, and M. D. G. V. Nunes, "A complex network approach to text summarization," *Inf. Sci.*, vol. 179, no. 5, pp. 584–599, Feb. 2009.
- [281] C. Kruengkrai and C. Jaruskulchai, "Generic text summarization using local and global properties of sentences," in *Proc. IEEE/WIC Int. Conf. Web Intell. (WI)*, Oct. 2003, pp. 201–206.
- [282] S. Kuppan and L. Sobha, "An approach to text summarization," in *Proc. 3rd Int. Workshop Cross Lingual Inf. Access, Addressing Inf. Need Multilingual Societies (CLIAWS3)*, 2009, pp. 53–60.
- [283] J. Tan, X. Wan, and J. Xiao, "Abstractive document summarization with a graph-based attentional neural model," in *Proc. 55th Annu. Meeting Assoc. Comput. Linguistics (Long Papers)*, vol. 1, 2017, pp. 1171–1181.
- [284] D. Suleiman and A. Awajan, "Deep learning based abstractive text summarization: Approaches, datasets, evaluation measures, and challenges," *Math. Problems Eng.*, vol. 2020, pp. 1–29, Aug. 2020.
- [285] L. Lebanoff, K. Song, and F. Liu, "Adapting the neural encoder-decoder framework from single to multi-document summarization," 2018, [arXiv:1808.06218](https://arxiv.org/abs/1808.06218).
- [286] R. Nallapati, B. Xiang, and B. Zhou, "Sequence-to-sequence RNNs for text summarization," in *Proc. ICLR*, Vienna, Austria, 2016.
- [287] S. Ma, X. Sun, J. Xu, H. Wang, W. Li, and Q. Su, "Improving semantic relevance for sequence-to-sequence learning of Chinese social media text summarization," 2017, [arXiv:1706.02459](https://arxiv.org/abs/1706.02459).
- [288] S. Syed, M. Völske, N. Lipka, B. Stein, H. Schütze, and M. Potthast, "Towards summarization for social media—results of the TL/DR challenge," in *Proc. 12th Int. Conf. Natural Lang. Gener.*, 2019, pp. 523–528.
- [289] T. Wang, P. Chen, K. Amaral, and J. Qiang, "An experimental study of LSTM encoder-decoder model for text simplification," 2016, [arXiv:1609.03663](https://arxiv.org/abs/1609.03663).
- [290] M. Tomer and M. Kumar, "Improving text summarization using ensemble approach based on fuzzy with LSTM," *Arabian J. Sci. Eng.*, vol. 45, no. 12, pp. 10743–10754, Dec. 2020.
- [291] Y. Diao, H. Lin, L. Yang, X. Fan, Y. Chu, D. Wu, D. Zhang, and K. Xu, "CRHASum: Extractive text summarization with contextualized representation hierarchical-attention summarization network," *Neural Comput. Appl.*, vol. 32, no. 15, pp. 11491–11503, Aug. 2020.
- [292] M. Tanaka and M. Okutomi, "A novel inference of a restricted Boltzmann machine," in *Proc. 22nd Int. Conf. Pattern Recognit.*, Aug. 2014, pp. 1526–1531.
- [293] N. Jaitly and G. Hinton, "Learning a better representation of speech soundwaves using restricted Boltzmann machines," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2011, pp. 5884–5887.
- [294] H. Larochelle, M. Mandel, R. Pascanu, and Y. Bengio, "Learning algorithms for the classification restricted Boltzmann machine," *J. Mach. Learn. Res.*, vol. 13, pp. 643–669, Mar. 2012.
- [295] C. L. P. Chen, C.-Y. Zhang, L. Chen, and M. Gan, "Fuzzy restricted Boltzmann machine for the enhancement of deep learning," *IEEE Trans. Fuzzy Syst.*, vol. 23, no. 6, pp. 2163–2173, Dec. 2015.
- [296] M.-A. Côté and H. Larochelle, "An infinite restricted Boltzmann machine," *Neural Comput.*, vol. 28, no. 7, pp. 1265–1288, 2016.
- [297] T. Kuremoto, S. Kimura, K. Kobayashi, and M. Obayashi, "Time series forecasting using restricted Boltzmann machine," in *Proc. Int. Conf. Intell. Comput.* Berlin, Germany: Springer, 2012, pp. 17–22.
- [298] P. Xie, Y. Deng, and E. Xing, "Diversifying restricted Boltzmann machine for document modeling," in *Proc. 21th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2015, pp. 1315–1324.
- [299] J. M. Conroy and D. P. O'leary, "Text summarization via hidden Markov models," in *Proc. 24th Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, 2001, pp. 406–407.
- [300] M. O. El-Haj and B. H. Hammo, "Evaluation of query-based Arabic text summarization system," in *Proc. Int. Conf. Natural Lang. Process. Knowl. Eng.*, Oct. 2008, pp. 1–7.
- [301] H. Jing, "Sentence reduction for automatic text summarization," in *Proc. 6th Conf. Appl. Natural Lang. Process.*, 2000, pp. 310–315.
- [302] H. Zha, "Generic summarization and keyphrase extraction using mutual reinforcement principle and sentence clustering," in *Proc. 25th Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retr. (SIGIR)*, 2002, pp. 113–120.
- [303] M. Kutlu, C. Cigir, and I. Cicekli, "Generic text summarization for Turkish," *Comput. J.*, vol. 53, no. 8, pp. 1315–1323, Oct. 2010.

- [304] I. Sorokin, A. Seleznev, M. Pavlov, A. Fedorov, and A. Ignateva, "Deep attention recurrent Q-network," 2015, *arXiv:1512.01693*.
- [305] G. H. Lee and K. J. Lee, "Automatic text summarization using reinforcement learning with embedding features," in *Proc. 8th Int. Joint Conf. Natural Lang. Process. (Short Papers)*, vol. 2, 2017, pp. 193–197.
- [306] A. P. Widyassari, A. Affandy, E. Noersasongko, A. Z. Fanani, A. Syukur, and R. S. Basuki, "Literature review of automatic text summarization: Research trend, dataset and method," in *Proc. Int. Conf. Inf. Commun. Technol. (ICOIAC)*, Jul. 2019, pp. 491–496.
- [307] Z. Deng, F. Ma, R. Lan, W. Huang, and X. Luo, "A two-stage Chinese text summarization algorithm using keyword information and adversarial learning," *Neurocomputing*, vol. 425, pp. 117–126, Feb. 2021.
- [308] M. E. Peters, M. Neumann, L. Zettlemoyer, and W.-T. Yih, "Dissecting contextual word embeddings: Architecture and representation," 2018, *arXiv:1808.08949*.
- [309] A. M. Dai and Q. V. Le, "Semi-supervised sequence learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 28, 2015, pp. 3079–3087.
- [310] S. R. Bowman, G. Angeli, C. Potts, and C. D. Manning, "A large annotated corpus for learning natural language inference," 2015, *arXiv:1508.05326*.
- [311] W. B. Dolan and C. Brockett, "Automatically constructing a corpus of sentential paraphrases," in *Proc. 3rd Int. Workshop Paraphrasing (IWP)*, 2005, pp. 1–8.
- [312] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 2018, *arXiv:1810.04805*.
- [313] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.
- [314] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.
- [315] V. Kieuvoongam, B. Tan, and Y. Niu, "Automatic text summarization of COVID-19 medical research articles using BERT and GPT-2," 2020, *arXiv:2006.01997*.
- [316] H. Froud, A. Lachkar, and S. A. Ouatik, "Arabic text summarization based on latent semantic analysis to enhance Arabic documents clustering," 2013, *arXiv:1302.1612*.
- [317] M. Mitra, A. Singhal, and C. Buckley, "Automatic text summarization by paragraph extraction," in *Intelligent Scalable Text Summarization*. Bangkok, Thailand: ACI Anthology, 1997.
- [318] A. R. Pal and D. Saha, "An approach to automatic text summarization using WordNet," in *Proc. IEEE Int. Advance Comput. Conf. (IACC)*, Feb. 2014, pp. 1169–1173.
- [319] S. Ryang and T. Abekawa, "Framework of automatic text summarization using reinforcement learning," in *Proc. Joint Conf. Empirical Methods Natural Lang. Process. Comput. Natural Lang. Learn.*, 2012, pp. 256–265.
- [320] H. G. Silber and K. F. McCoy, "Efficiently computed lexical chains as an intermediate representation for automatic text summarization," *Comput. Linguistics*, vol. 28, no. 4, pp. 487–496, Dec. 2002.
- [321] F. Chua and S. Asur, "Automatic summarization of events from social media," in *Proc. Int. AAAI Conf. Web Social Media*, vol. 7, no. 1, 2013, pp. 1–10.
- [322] J. Xu, Z. Gan, Y. Cheng, and J. Liu, "Discourse-aware neural extractive text summarization," 2019, *arXiv:1910.14142*.
- [323] R. M. Aliguliyev, "A new sentence similarity measure and sentence based extractive technique for automatic text summarization," *Expert Syst. Appl.*, vol. 36, no. 4, pp. 7764–7772, May 2009.
- [324] R. Aliguliyev and R. Aliguliyev, "Evolutionary algorithm for extractive text summarization," *Intell. Inf. Manage.*, vol. 1, no. 2, p. 128, 2009.
- [325] R. Ferreira, L. de Souza Cabral, R. D. Lins, G. P. e Silva, F. Freitas, G. D. C. Cavalcanti, R. Lima, S. J. Simske, and L. Favaro, "Assessing sentence scoring techniques for extractive text summarization," *Expert Syst. Appl.*, vol. 40, no. 14, pp. 5755–5764, Oct. 2013.
- [326] C. Fang, D. Mu, Z. Deng, and Z. Wu, "Word-sentence co-ranking for automatic extractive text summarization," *Expert Syst. Appl.*, vol. 72, pp. 189–195, Apr. 2017.
- [327] J. Xu and G. Durrett, "Neural extractive text summarization with syntactic compression," 2019, *arXiv:1902.00863*.
- [328] T. K. Landauer, P. W. Foltz, and D. Laham, "An introduction to latent semantic analysis," *Discourse Process.*, vol. 25, nos. 2–3, pp. 259–284, 1998.
- [329] Y. Kikuchi, T. Hirao, H. Takamura, M. Okumura, and M. Nagata, "Single document summarization based on nested tree structure," in *Proc. 52nd Annu. Meeting Assoc. Comput. Linguistics (Short Papers)*, vol. 2, 2014, pp. 315–320.
- [330] L. Liu, Y. Lu, M. Yang, Q. Qu, J. Zhu, and H. Li, "Generative adversarial network for abstractive text summarization," in *Proc. AAAI Conf. Artif. Intell.*, vol. 32, no. 1, 2018, pp. 1–3.
- [331] Y. Ledeneva, A. Gelbukh, and R. A. García-Hernández, "Terms derived from frequent sequences for extractive text summarization," in *Proc. Int. Conf. Intell. Text Process. Comput. Linguistics*. Berlin, Germany: Springer, 2008, pp. 593–604.
- [332] Y.-C. Chen and M. Bansal, "Fast abstractive summarization with reinforce-selected sentence rewriting," 2018, *arXiv:1805.11080*.
- [333] R. Barzilay and M. Elhadad, "Using lexical chains for text summarization," in *Advances in Automatic Text Summarization*. Cambridge, MA, USA: MIT Press, 1999, pp. 111–121.
- [334] E. Hovy and C. Y. Lin, "Automated text summarization in SUMMARIST," *Adv. Autom. Text Summarization*, vol. 14, pp. 81–94, Oct. 1999.
- [335] P. W. Foos, "The effect of variations in text summarization opportunities on test performance," *J. Experim. Educ.*, vol. 63, no. 2, pp. 89–95, Jan. 1995.
- [336] R. Shardan and U. Kulkarni, "Implementation and evaluation of evolutionary connectionist approaches to automated text summarization," 2010.
- [337] G. Padmapriya and K. Duraiswamy, "An approach for text summarization using deep learning algorithm," *J. Comput. Sci.*, vol. 10, no. 1, pp. 1–9, Jan. 2014.
- [338] N. Yadav, "Neighborhood rough set based multi-document summarization," 2021, *arXiv:2106.07338*.
- [339] M. Yang, C. Li, Y. Shen, Q. Wu, Z. Zhao, and X. Chen, "Hierarchical human-like deep neural networks for abstractive text summarization," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 6, pp. 2744–2757, Jun. 2021.
- [340] B. Elayeb, A. Chouigui, M. Bounhas, and O. B. Khiroun, "Automatic Arabic text summarization using analogical proportions," *Cognit. Comput.*, vol. 12, no. 5, pp. 1043–1069, Sep. 2020.
- [341] W. S. El-Kassas, C. R. Salama, A. A. Rafea, and H. K. Mohamed, "Automatic text summarization: A comprehensive survey," *Expert Syst. Appl.*, vol. 165, Mar. 2021, Art. no. 113679.
- [342] D. R. Radev and D. Tam, "Summarization evaluation using relative utility," in *Proc. 12th Int. Conf. Inf. Knowl. Manage. (CIKM)*, 2003, pp. 508–511.
- [343] S. Teufel and H. V. Halteren, "Evaluating information content by Factoid analysis: Human annotation and stability," in *Proc. ACL*, Bangkok, Thailand, 2004.
- [344] L. Alonso I Alemany and M. Fuentes Fort, "Integrating cohesion and coherence for automatic summarization," in *Proc. 10th Conf. Eur. Chapter Assoc. Comput. Linguistics (EACL)*, 2003, pp. 1–8.
- [345] I. Mani, E. Bloedorn, and B. Gates, "Using cohesion and coherence models for text summarization," in *Proc. Intell. Text Summarization Symp.*, 1998, pp. 69–76.
- [346] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: A method for automatic evaluation of machine translation," in *Proc. 40th Annu. Meeting Assoc. Comput. Linguistics (ACL)*, 2001, pp. 311–318.
- [347] Y. Graham, "Re-evaluating automatic summarization with BLEU and 192 shades of Rouge," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2015, pp. 128–137.
- [348] M. Popović, "ChrF: Character n-gram F-score for automatic MT evaluation," in *Proc. 10th Workshop Stat. Mach. Transl.*, 2015, pp. 392–395.
- [349] O. Bojar, Y. Graham, A. Kamran, and M. Stanojević, "Results of the WMT16 metrics shared task," in *Proc. 1st Conf. Mach. Transl., Shared Task Papers*, vol. 2, 2016, pp. 199–231.
- [350] Q. Ma, O. Bojar, and Y. Graham, "Results of the WMT18 metrics shared task: Both characters and embeddings achieve good performance," in *Proc. 3rd Conf. Mach. Transl., Shared Task Papers*, 2018, pp. 671–688.
- [351] D. R. Radev, S. Teufel, H. Saggion, W. Lam, J. Blitzer, H. Qi, A. Çelebi, D. Liu, and E. Drabek, "Evaluation challenges in large-scale document summarization," in *Proc. 41st Annu. Meeting Assoc. Comput. Linguistics (ACL)*, 2003, pp. 375–382.
- [352] P. Malik, A. Gupta, and A. Baghel, "Key issues in machine translation evaluation of English-Indian languages," *Int. J. Eng. Res. Technol.*, vol. 2, no. 10, p. 2278, 2013.
- [353] B. Babych, "Automated MT evaluation metrics and their limitations," *Tradumática*, no. 12, pp. 464–470, Dec. 2014.



M. F. MRIDHA (Senior Member, IEEE) received the Ph.D. degree in AI/ML from Jahangirnagar University, in 2017. He joined the Department of Computer Science and Engineering, Stamford University Bangladesh, as a Lecturer, in June 2007. He was promoted as a Senior Lecturer at the Department of Computer Science and Engineering, in October 2010, and promoted as an Assistant Professor at the Department of Computer Science and Engineering, in October 2011. Then he joined UAP, as an Assistant Professor, in May 2012. He also worked as a Faculty Member at the CSE Department, University of Asia Pacific, and as a Graduate Co-ordinator, from 2012 to 2019. He is currently working as an Associate Professor with the Department of Computer Science and Engineering, Bangladesh University of Business and Technology. For more than ten years, he has been with the master's and bachelor's students as a supervisor of their thesis work. His research experience, within both academia and industry, results in over 80 journal and conference publications. His research interests include artificial intelligence (AI), machine learning, deep learning, big data analysis, and natural language processing (NLP). He has served as a program committee member for several international conferences/workshops. He served as an associate editor for several journals.



AKLIMA AKTER LIMA is currently pursuing the degree in computer science with the Bangladesh University of Business and Technology. She is a well-organized, determined to work. She is also working as an Assistant Researcher with the Advanced Machine Learning Laboratory. She has experience working with Tensorflow, Keras, and Matplotlib. Her research interests include machine learning and deep learning. She is also researching advanced driver assistance systems, stock exchange, and automatic text summarization.



KAMRUDDIN NUR (Senior Member, IEEE) received the bachelor's degree from Victoria University of Wellington (VUW), New Zealand, the master's degree from UIU, and the Ph.D. degree from UPF, Barcelona, Spain. He is currently working as an Associate Professor with the Department of Computer Science, American International University-Bangladesh (AIUB). He also served as the Chairperson with the Department of Computer Science and Engineering, Stamford University Bangladesh (SUB); and the Bangladesh University of Business and Technology (BUBT). He authored many prestigious journals and conferences, such as IEEE and ACM. His research interests include ubiquitous computing, computer vision, machine learning, and robotic automation. He served as a TPC member, and reviewed articles in IEEE, ACM, Springer journals, and conferences.



SUJOY CHANDRA DAS is currently pursuing the degree in computer science with the Bangladesh University of Business and Technology. He is a determinant, communicative, and sincere to work. He is also working as an Assistant Researcher with the Advanced Machine Learning Laboratory. He has good communication and presentation skills. He has experience working with front-end development, Tensorflow, Keras, and Matplotlib. His research interest includes deep learning. He is also researching the advanced driver assistance systems and automatic text summarization.



MAHMUD HASAN received the bachelor's degree in computer science from the Chittagong University of Engineering and Technology, in 2011, and the M.Sc. degree in computer science from Western University, Canada, in 2014, where he is currently pursuing the Ph.D. degree with the Department of Computer Science. He started his professional career as a Lecturer at Stamford University Bangladesh. He later joined the Bangladesh University of Textiles, as a Lecturer. He worked as a Research Software Developer at the Robarts Research Institute, as a Staff Software Developer with the Research and Development Unit, IBM Watson Health Imaging, and as a Principal Investigator at Acceo Tender Retail. He is also working as a Research Analyst at the Department of Medical Biophysics, Western University. In his versatile career moves, he closely worked with digital and medical image analysis, understanding, segmentation, registration, compression, classification, denoising, and computer vision. He has proved track record of publications in quality journals and talks presented in many conferences. His primary focus is on biomedical image analysis using deep learning. His research interests include other application of areas of machine learning in general, such as NLP.



MUHAMMAD MOHSIN KABIR was born in Dhaka, Bangladesh. He received the B.Sc. degree in computer science and engineering from the Bangladesh University of Business and Technology (BUBT). He is currently working as a Lecturer and a Research Assistant with the Department of CSE, BUBT. He also works as a Researcher with the Advanced Machine Learning Laboratory. He is always willing to learn new things with full enthusiasm and passion. He has experienced working in Python, Keras, TensorFlow, Sklearn, and Scipy. His research interests include deep learning, pattern recognition, computer vision, and natural language processing.

...