Leveraging DistilBERT for Summarizing Arabic Text: An Extractive Dual-Stage Approach

ABDULLAH ALSHANQITI1, ABDALLAH NAMOUN1, (Member, IEEE), AESHAH ALSUGHAYYIR2, AISHA MOUSA MASHRAQI3, ABDUL REHMAN GILAL4, AND SAMI SAAD ALBOUQ1

1Faculty of Computer and Information Systems, Islamic University of Madinah, Madinah 42351, Saudi Arabia
2College of Computer Science and Engineering, Taibah University, Madinah 42353, Saudi Arabia
3College of Computer Science, Najran University (NU), Najran 61441, Saudi Arabia
4Department of Computer Science, Sukkur IBA University, Sukkur 65101, Pakistan

Corresponding author: Abdullah Alshanqiti (amma@iu.edu.sa)

This work was supported by the Deanship of Scientific Research, Islamic University, Madinah, under Takamol Project 11_209_2021.

ABSTRACT Towards tackling the phenomenon of textual information overload that is exponentially pumping with redundancy over the Internet, this paper investigates a solution depending on the Automatic Text Summarization (ATS) method. The idea of ATS is to assist, e.g., online readers, in getting a simplified version of texts for preserving their time/effort required to skim a given large body of text. However, ATS is deemed as one of the most complex NLP applications, particularly for the Arabic language that has not been intelligently developed like the other Indo-European languages. Thus, we present an extractive-based summarizer (ArDBertSum) for text written in Arabic, relying on the DistilBERT model. Besides, we propose a domain-specific sentence-clauses segmenter (SCSAR) to support our ArDBertSum in further shortening long/complex sentences. The results of our experiments illustrate that our ArDBertSum yields the best performance, compared with non-heuristic Arabic summarizers, in producing an acceptable quality of candidate summaries. These experiments have been conducted on EASC-dataset (along with our proposed dataset) to report on (1) a statistical evaluation utilizing ROUGE metrics and (2) a specific human-based evaluation. The human evaluation results revealed promising perceptions; however, further works are needed to ameliorate the coherence and punctuation of the automatic summaries.

INDEX TERMS NLP, natural language understanding, automatic text summarization, machine learning, transfer learning, pre-trained model, BERT, sentence segmentation, clause-based segmentation, sentence reduction.

I. INTRODUCTION

In this era of digitalization, tremendous amounts of textual data and electronic documents are exponentially pumping and advance to diffuse over the Internet rapidly. Habitually, online users found it time-consuming and effortful to find the desired information within a large body of text, particularly when filtering out the retrieved information using current search engines. This consideration has motivated many researchers to contribute to Automatic Text Summarization (ATS) within the Natural Language Processing (NLP) for saving user’s effort and time [1]. Here, ATS technologies can be of magnificent help in addressing the information overload problem by, e.g., suggesting solutions that work around shortening long texts with summaries without losing the intended meaning [2].

Although ATS has been researched for approximately 60 years now [3], it still remains a very active research area as many problems related to text analysis and semantic complexities have not been solved entirely yet. Glancing over the ATS literature, one can observe that most intelligent studies and summarizing methodologies have been conducted to support the Indo-European languages, such as English, while limited works have been dedicated to supporting the Arabic language. While the Arabic language is a fast-growing language on the Internet with an annual growth rate of 9,348.0% (measured between 2000 - 2021),1 making it the fourth most spoken language in the world, it is less explored with respect

1https://www.internetworldstats.com/stats7.htm
to ATS due to its complex syntax, structure, and verb conjugation [4]. Additionally, most Arabic-ATS approaches are experiencing poor to moderate performance [2] due to the high linguistic complexity of the Arabic language itself. To clarify more, considering the low-resource Arabic NLP, the available supporting tools are suffering from many difficulties [5]. Few of these many difficulties include (1) ambiguity in recognizing the POS-tagging correctly in the absence of diacritics, which is often the case; (2) ambiguity in identifying word-lematization (e.g., removing inflectional endings correctly), which is because the Arabic language has a rich derivational morphology; and (3) lack of having necessary parsing and tokenizing resources, including comprehensive lexicons.

Furthermore, to the extent of our knowledge, few extractive-based techniques exist in Arabic summarization, and no work is suggested based on abstractive technique. The extractive technique means to shorten a given textual documents by only focusing on essential sentences presented in the original documents and ignore less important ones. The abstractive technique, however, can generate new texts that may not exist in the original documents. Broadly, there are different extractive-based techniques through which Arabic ATS is achieved by many researchers, e.g., [5]–[15]. These techniques can be categorized into different approaches: the statistical approach, the semantic approach, the machine learning approach, and the meta-heuristic approach. In a little more detail, Al-Abdallah and Al-Taani [9] discuss the meta-heuristic approach in Arabic summarization and used Particle Swarm Optimization (PSO) algorithm to obtain extractive Arabic text summarization. Azmi and Al-Thanyyan [6] used Rhetorical Structure Theory (RST) that is a statistical approach to form a candidate summary based on RST between the terms. In order to group the similar sentences statistically, [13] used a clustering method to extract the candidate summary by selecting the most important sentences in the clusters. The study [16] presents the machine learning approach by using the Adaboost algorithm for Arabic text summarization. We will give more discussion on the closely related approaches in the next section.

 Seeking to investigate the capability of the deep transfer learning model to tackle the Arabic ATS problem, this paper presents a text summarization system based on a Distilled Bidirectional Encoder Representations from Transformers model (so-called DistilBERT [17]). This model is one of the emerging state-of-the-art pre-trained language understanding models, distinguished by its lightness and efficiency compared to the large (base) BERT [18]. In a nutshell, this paper introduces the following contributions:

- We introduce an Arabic summarizer approach (ArD-BertSum) that consolidates two extractive summarization stages, built upon DistilBERT. The second supporter stage strives to enhance the typical summarization method used (i.e., depends on sentence-based selection) by further shorting long sentences.

- We propose a domain-specific sentence-clauses segmentation for Arabic texts (SCSAR).
- We make a new Arabic corpus available online for benchmarking human-based summarization approaches.
- Furthermore, we report on an implementation and experimental evaluation (including a user-based experiment), highlighting the efficient performance and practicality of our proposal. The implementation and coding details are publicly accessible to the interested researchers for replicating our experiments.

In the following sections, we first review gaps in the literature concerning the Arabic ATS approaches and present noteworthy limitations of the existing techniques. Next, we introduce our proposal in detail that is Arabic-domain-specific, and then present the subsequent experimental analysis and findings. Summing up the paper and outlining the potential future avenues of research are given afterward.

### II. RELATED LITERATURE

A quite long history of ATS approaches and methodologies exists in the NLP literature, probably since the 1950s [3]; many of them have been recently surveyed in [1]. Therefore, to shed some light on the closely related works only, this section primarily discusses the body of work in the Arabic ATS literature (cf. Section II-A) and reviews some of their common shortcomings (cf. Section II-B). To broadly simplify the underlying process that describes how summarization methods work, we mention the primary three relatively common steps involved in almost all summarizers: (1) Structure (with encoding if applicable) a proper text representation of the given original text. (2) Extract the required features from the created text representation to, e.g., score sentences. (3) Concatenate the important sentences according to the selection criteria to generate a candidate summary, relying on the extracted features (mainly, i.e., the weighted score of sentences).

### A. OVERVIEW OF ATS APPROACHES

The earliest works on ATS have concentrated on manipulating the word frequency to form statistical candidate summaries [3], and their problematic concerns were on how to produce a readable summary [19], [20]. With the continued progress of research in addressing ATS, reaching the 2020s, many different approaches have been suggested, which can be broadly categorized [1], [5], [13] as described in Table 1.

### Table 1. A brief categorization of ATS methodologies.

| Method | Language | Context’s Generality | Input | Output |
|--------|----------|----------------------|-------|--------|
| Extractive | Monolingual | Domain-specific | Single-document | Indicative |
| Abstractive | Multilingual | Generic | Multi-document | Informative |
| Hybrid | | Query-based Driven | | |

2 https://github.com/AMahfodh/ArSummarizer
This paper studies a generic and extractive-based approach for the Arabic language that seeks to output an informative candidate summary from a given single document. More in details, the abstractive summarization method aims to construct new sentences (sometimes with paraphrasing technique [21]) to produce a candidate summary, relying on understanding the observed input texts, see [22]. While the extractive method weights and selects sentences from the input texts based on ranking criteria. The latter method is popular (i.e., especially in news domain [23]) and seems a more straightforward method, which tends to produce a higher efficiency summary than the abstractive-based summary [24].

Diving into the extractive method that we consider in this paper, there are various techniques used in addressing Arabic summarization, such as statistical/linguistic-based [6], [8], [9], [15], semantic/query-based [7]–[9], [12], [21], graph/optimization-based [10]–[12], [25], and machine learning [5], [13]–[15], [19], [20], [26]–[28]. For a detailed review of these techniques for the Arabic ATS, we recommend that researchers refer to these recent works [1], [5], [8]. However, in Table 2, we summarize the text summarization techniques reviewed in this paper, and in Table 3, we briefly describe the Arabic summarization corpora.

In statistical technique, ranking to select the estimated important sentences and words from the input text often depends on the ability to extract statistical and linguistic features (e.g., most frequent tokens or marked part-of-speech tagging) within the whole document [24]. For example, [25] proposed a topic aspect-oriented summarization (TAOS), which extracts the following features from a given multi-document: sentence length, sentence position, and word frequency and then applies a greedy algorithm for scoring and selecting sentences.

For the Arabic ATS, [9] proposed an extractive-summarizer implemented using Particle Swarm Optimization (PSO) algorithm. Their summarizer combines the advantages of statistical and semantic-based techniques. A graph-based ranking model is examined recently in [12] with the purpose of addressing a specific case of ATS (i.e., update-summarization). The authors here introduced the first novel approach for tackling the Arabic update-summarization problem. An extractive query-based text summarization model is explored in [7]. Here, the authors apply the Latent Semantic Analysis technique by exploiting the Arabic WordNet (AWN) ontology to produce a candidate summary semantically. Moreover, the Rhetorical Structure Theory is explored in [6], with an attempt at identifying the rhetorical relationships between token-terms, illustrated in a tree form. Here, the candidate summary is produced by firstly selecting a specific set of rhetorical relations followed by selecting their mapped sentences.

Various researchers in the machine learning communities have studied ATS problem with different learning techniques, including e.g., classical supervised learning [16], [19], unsupervised clustering-based learning [5], [10], [13], [15], reinforcement-based learning [20], and deep learning based on artificial neural networks [14], [18], [26]–[36]. For instance, [16] developed an Arabic text summarizer based on the adaptive boosting model (a.k.a, AdaBoost, which is supervised-based learning used typically to optimize the weak decision-tree classifiers). Concerning the unsupervised learning technique, there has been recently rising concern in exploring the clustering methods. For instance, [13] design an Arabic text summarizer that focuses on reducing the redundancy and noisy data in a given input multi-document. Their underlying technique is implemented using an unsupervised score-based method. Similarly, [15] proposed an extractive cluster-based summarizer using the minimum redundancy and the maximum relevance (mRMR) strategy for scoring and identifying the important sentences.

The artificial neural network models have been widely used in tackling many NLP problems. However, a few deep-learning techniques have been suggested for summarizing texts written in Arabic, including [14], [26]–[28]. Here, Alami et al. [14] propose unsupervised deep learning for Arabic ATS based on variational auto-encoder (VAE) to optimize the representation of input text. While we consider the extractive summarization method, [27] focused on generating an abstractive headline from the introduction of an article, relying on an encoder-decoder framework, analogous to the method presented in [29]. Generally speaking, deep learning techniques have achieved state-of-the-art results in various Natural Language Understanding (NLU) tasks, depending on transfer learning and/or adopting pre-trained models [32], [33]. Recently, many leading research teams have offered various pre-trained models for the NLP communities to fine-tune on different downstream tasks, including the Google Brain-team [31], [34] (i.e., propose an abstractive sequence-to-sequence with attention model for summarizing textual news) and Facebook AI-team [29] (i.e., propose an attention-based summarization (ABS) model for abstractive text summarization).

The current state of the art pre-training models for language understanding problem includes, e.g., GPT [35], XLNet [36], BERT [18], [30], and many others. Most of the publicly available pre-trained models were not fine-tuned explicitly for the Arabic ATS tasks to the best of the authors’ knowledge. Nonetheless, the authors of [26], [28] have attempted to utilize BERT for the Arabic ATS tasks. In specific, [28] merely experimented on two existing models (BertSUM and BertSUMAbs) for summarizing the Arabic texts. These models are originally designed by [30] for English texts. While [26] utilized the AraBERT model for sentence embedding only, then they use the K-Means algorithm for clustering the generated embedding. In line with [26], [28] but under different procedures, this paper introduces a subsequent attempt to investigate the utility of monolingual pre-trained NLU models (i.e., a distilled version of BERT [18]) for tackling the Arabic ATS problem in a more comprehensive investigation.
### Table 2. A brief summary of reviewed text summarization techniques.

| Genre                        | Example                                                                 | Comparison criteria | Language | Summarization type | Input | Corpus        | Metric                         |
|------------------------------|-------------------------------------------------------------------------|---------------------|----------|--------------------|-------|---------------|--------------------------------|
| Meta-heuristic               | [9] studies the use of the Particle Swarm Optimization (PSO) algorithm for the ATS problem. |                     | Ar       | Ext                | Single | EASC          | ROUGE                          |
| Statistical                  | [25] studies a topic aspect-oriented summarization method (TAOS)         |                     | En       | Ext                | Multi  | DUC           | Jensen Shannon Divergence & ROUGE |
| Rhetorical structure theory  | [6] studies Rhetorical Structure Theory (RST), that is a statistical approach, to form a candidate summary based on RST between the terms. |                     | Ar       | Ext                | Single | Authors corpus| ROUGE                          |
| Query and semantic-based     | [7] studies a query-based technique with the latent semantic analysis, focusing on the Arabic WordNet (AWN) ontology, to produce a candidate summary semantically. Similarly, [8] developed a hybrid model based on statistical and semantic treatment. |                     | Ar       | Ext                | Single | EASC & Authors corpus | ROUGE                          |
| Graph-based                  | [12] proposes a query-based update-summarization System (AUSS) using a graph-based ranking model. |                     | Ar       | Ext                | Multi  | Authors corpus (AUS-DB) | ROUGE                          |
| Classical machine learning   | [16] developed an Arabic text summarizer based on the supervised AdaBoost model. |                     | Ar       | Ext                | Single | Authors corpus | Precision, Recall & F-measure | ROUGE                          |
| Artificial neural network    | [13] developed a clustering-based model to extract the candidate summary, focusing on reducing the redundancy and noisy data in a given input multi-document. [10] proposed a similar method but based on the A* algorithm. |                     | Ar       | Ext                | Multi  | EASC          | ROUGE                          |
|                             | [15] proposes an extractive cluster-based summarizer using the minimum redundancy and the maximum relevance strategy for scoring and identifying the important sentences. |                     | Ar       | Ext                | Single & Multi | MultiLing Pilot 2011 & EASC | ROUGE                          |
|                             | [14] proposes unsupervised deep learning for Arabic ATS based on variational auto-encoder (VAE) to optimize the representation of input text. |                     | Ar       | Ext                | Single | EASC & Authors corpus | ROUGE                          |
|                             | [27] focused on generating an abstractive headline from the introduction of an article, relying on an encoder-decoder framework, analogous to the method presented in [29]. |                     | Ar & En  | Hyb                | Single | Authors corpus (AHS) | ROUGE                          |
|                             | [28] studies two existing models (BertSUM and Bert-SUMAhs) for summarizing the Arabic texts. These models are originally designed by [30] for English texts. |                     | Ar       | Abs & Ext          | Single | KALIMAT       | ROUGE                          |
|                             | [26] utilized the AraBERT model for sentence embedding only; then, they used the K-Means algorithm for clustering the generated embedding. |                     | Ar       | Ext                | Single | Authors corpus | ROUGE                          |
|                             | [23] studies seq2seq model based on LSTM and attention mechanisms to generate a candidate summary. |                     | En       | Abs                | Single | Gigaword      | ROUGE                          |
|                             | [29] studies a local attention-based model that generates each word of the summary conditioned on the input sentence. |                     | En       | Abs                | Single | DUC           | ROUGE                          |
|                             | [31] attempted to address the ATS problem using Pointer-Generator Networks, depending on the sequence-to-sequence model with attention. |                     | En       | Abs                | Single | CNN/Daily Mail | ROUGE                          |
|                             | Our proposed ArDBertSum that supports shortening long/complex sentences. It’s built upon DistilBERT model. |                     | Ar       | Ext                | Single | EASC & Authors corpus | ROUGE                          |

Comparison criteria:
- **Language:** Arabic | English
- **Summarization method:** Abstractive | Extractive | Hybrid
- **SBS:** Sentence-based Selection
- **SBR:** Sentence-based Reduction
- **Input type:** Single-document | Multi-document
### TABLE 3. A description of Arabic summarization corpora.

| Corpus | Language | Numbers | Input Type | Gold-standard summaries |
|--------|----------|---------|------------|------------------------|
| EASC (see Table 5) | Arabic | 153 | Single document | 5 summaries per document |
| AUS-DB | Arabic | 183 | Multi document | 22 summaries per topic (10 documents) |
| MultiLing 2013 | Arabic | 30 | Single document | 1 summary per document |
| | Arabic | 150 | Multi document | 3 summaries per topic (15 documents) |
| KALIMAT Text Analysis Conference (TAC) MultiLing 2011 Document Understanding Conference (DUC) | Arabic | 20291 | Single document | 1 summary per document |
| | Arabic | 880 | Multi document | 3 summaries per topic (10 documents) |
| | Arabic | 1000 | Single and Multi documents | 10 summaries per document |

### B. LIMITATIONS OF REVIEWED APPROACHES

ATS is a complex NLP task, especially with the Arabic language that has an extremely rich derivational morphology. From the recent literature we reviewed, there are many unresolved common challenges in ATS, such as criteria for stopping text generation, the quality as well as and the evaluation of the generated candidate summaries [5]. Nonetheless, in this paper, we are mainly focusing on the quality aspects as follows:

- Improve sentence-based segmentation at the complex clause-based level to further shorten long sentences independently; and
- Investigate the overall utility of DistilBERT model in Arabic summarization using some statistical ROUGE metrics. This investigation would implicitly estimate the efficiency of the input intermediate-representation technique based on DistilBERT for the Arabic texts (i.e., depends on the word-embedding method) as well as the accuracy of sentence tokenization and scoring.

### III. ArDBertSum: PROPOSED DistilBERT-BASED SUMMARIZER

In this section, we detail our summarization method ArDBertSum that is dedicated to the Arabic text. The underlying structure of ArDBertSum is built upon DistilBERT [17] and represented as a sentence-based selection problem. Let \( X \) denote an input document, which consists of a sequence of sentences \( X = \{x_1, x_2, \ldots, x_n\} \), and use \( Y = \{y_1, y_2, \ldots, y_m\} \) to denote the generated candidate summary, such that \( m \leq n \). Here, with the aim of generating a precise \( Y \), the proposed ArDBertSum sustains two distinct text preprocessing procedures; each is performed prior to using DistilBERT for selecting sentences (followed by also selecting their fragmented clauses). We illustrate the overall flowchart of ArDBertSum in Figure 1, and explain its stages in the next subsections.

### A. TEXT PREPROCESSING

Intuitively, text preprocessing is a vital step for cleaning/preparing textual documents, which is often performed differently within NLP approaches. Therefore, to lay out our preprocessing method, we consider three high-level NLP-components: normalization, noise removal, including the Arabic diacritizations, and segmentation (i.e., includes sentence-based tokenization). These components have a positive effect in terms of accuracy (represented by, e.g., ROUCH Score), and most importantly, they maintain the overall meaning to a reasonable extent. In this paper, we meant to neglect all preprocessing components that could break sentence structure (e.g., removal of stop words) in order to produce a readable and concise candidate summary after the preprocessing. Figure 2 gives a self-explanatory example with an input \( X \) that illustrates the purpose and effect of applying the preprocessing components considered in this paper.

More in detail concerning the back-end implementation, we adopt two Python-based toolkits: NLTK\(^3\) and CAMeL.\(^4\) Both toolkits are very rich and powerful, but the latter focuses more on the Arabic texts, allowing to perform many specific analysis tasks, such as parsing, tokenization, and stemming, ... etc. Despite the richness of such toolkits, identifying the right boundaries of sentences (and their clauses) segmentations is a problematic and not a straightforward task. This unavailable and such needed feature has prompted us to contribute with a particular sentence clause-based segmenter for the Arabic texts (SCSAR), discussed in the next subsection.

### B. DistilBERT FOR TEXT REPRESENTATION AND SUMMARIZATION

Inspired by BERT (Bidirectional Encoder Representations from Transformers) [18] that has gained superior improve-
FIGURE 1. The proposed DistilBERT-based method consisting of two summarization stages.

Conceptually speaking, DistilBERT is trained using reinforcement learning, relying on the teacher-student network model, to produce (1) a much lower dimensional model compared to BERT (i.e., smaller) and (2) a ready-trained model based on knowledge compression and transformation [39]. This means the hidden kernel layers of DistilBERT (i.e., representing a small student network model $M^s$) do not necessarily require further datasets to re-train, but rather it gains distilled knowledge from the big BERT model (in this case the teacher network $M^t$) that is previously trained on a massive amount of data for a considerable time. The distillation and the transformation of such knowledge (i.e., from $M^t$ to $M^s$) are performed based on a reinforcement technique [40] with a particular activation function added in both networks’ output layers (i.e., softmax temperature) and under a loss objective function expressed in Eq. (1):

$$L_{ins}(M^s; M^t) = \alpha \ast H(l_{out}, \sigma(M^s_{out}; T = 1)) \quad + \beta \ast H(\sigma(M^t_{out}; T = \tau), \sigma(M^s_{out}, T = \tau) |l|)$$

where $ins$ is the input instance; $l_{out}$ is the ground truth label; $H$ is the cross-entropy loss function; $\sigma$ is the softmax activation function parameterized by the temperature $T$; $\alpha$ and $\beta$ are the hyper coefficient-parameters; $M^t_{out}$ and $M^s_{out}$ are the outputs of teacher and student models obtained by their softmax function.

In our work, we utilize a fine-tuned version of DistilBERT for the generic summarization task ($dB_{sum}$), implemented by the Hugging Face. Under the hood, this generic summarizer $dB_{sum}$ does three consecutive tasks: (1) representing a given input document $X$ based on different embeddings; (2) tokenizing $X$ into sentences such that each sentence begins and ends with specific tokens (i.e., [CLS] and [SEP] respectively); and (3) selecting sentences to be composed as a candidate summary based on their mapped outputs with [CLS] token. To clarify more, we abstractly express the process of $dB_{sum}$ as follows:

$$dB_{sum}(X, \theta) = \sum_{s=1}^{n} (#DistilBERT_{output}(x_s))$$

if $\gamma(x_s^{[CLS]}) > \theta$  

where $n$ is the number of tokenized sentences; $#$ represents the concatenation of selected sentences; $\theta$ is a hyper
threshold-parameter that specifies the allowed confident prediction for composing \(s_i\) in the generated candidate summary; and \(\gamma(s_i^{[CLS]})\) represents the logits after the transformer layer that is mapped with the token [CLS].

C. SCSAR: SENTENCE-CLAUDES SEGMENTATION FOR ARABIC TEXTS

As illustrated in Figure 1(B), the candidate summary \(Y\) is produced by DistilBERT after segmenting the generated initial candidate summary \(Y_{ini}\) into clauses (i.e., detected clauses are re-separated by a full-stop punctuation mark as a re-preprocessing step, see the bottom part of Figure 2). In this case, referring to (2), our eventual candidate summary can be expressed as \(Y = d_{sum}(Y_{ini}, \theta)\), where \(Y_{ini} = d_{sum}(X, \theta)\). Predominantly, chunking long-complex sentences that may present in \(Y_{ini}\) into independent clauses is a significant step in enhancing the quality of \(Y\) as it helps the summarizer \(d_{sum}(. . .)\) in detecting and excluding the less essential clauses during summarization process. To our knowledge, as yet, there is no definitive solution to address the detection of clause segments for the Arabic text fundamentally. The most promising but highly complex approach suggests digging deeper and identifying the syntactic function of sentences (subject, object, verb, etc.), including the detection of their shared linguistic typology. Nevertheless, such an approach still suffers from conjunction ambiguity at the treebank level [41].

From a more general linguistic outlook, we propose (SCSAR) a practical pre-processing rule solution, relying on different types of implicit conjunctions (apart from the apparent punctuation marks, such as (‘,’ ‘;’ ‘?’)), for detecting clause splits. In Table 4, we present a comprehensive list containing almost all sorts of conjunctions used typically with formal Arabic writing. For simplicity, we consider two high-level types of clauses: independent or dependent (i.e., subordinate) with assuming that any complex sentence must include at least one independent (main) clause with coordinating conjunction(s). The top-left part of Table 4 describes six possible permutated cases between these types of clauses. Here, we have investigated the potential occurrence of these cases (i.e., A-F) in complex sentences by the collected 70 conjunctions and estimated the ambiguity rate for each conjunction. The result realized by this investigation, summed up in Table 4, has been reviewed by an expert consultant in Arabic linguistics. To enhance the quality of the produced candidate summary, domain-expert are flexible to include/exclude the appropriate conjunctions, taking into account their estimated ambiguity rate.

More in detail for implementing our SCSAR, we use a custom rule-based strategy available in SpaCy,\(^7\) which supports an optimized feature for pre-defining segment boundaries in advance (i.e., prior to the preprocessing step). The overall procedure of SCSAR is described in algorithm 1, which preprocesses a given document in a way that enables text tokenization task to be performed at the independent sentence-clauses level. In particular, sentences are chunked into potentially independent clauses whenever (1) the pre-defined conjunctions are observed and (2) the nominated clauses are estimated to be valid (i.e., either noun clause or main clause containing at least one subject and one verb).

IV. EXPERIMENTS AND KEY FINDINGS

By glancing over the many and various approaches available for addressing text summarization problems, one can deduce that summarization’s quality/precision is highly subjective and often affected by contrasting human understandings. Even with the use of statistical benchmarking methods, the challenge remains as the preparation steps (e.g., datasets cleaning, encoding and normalization) always contrast strikingly within the proposed approaches. Consequently, it is quite challenging to place all related approaches in a single standardized form for comparison. Nevertheless, we carried out several experiments to estimate the effectiveness of our summarizer (including the proposed SCSAR) from two perspectives:

- statistical perspective that involves a comparison with the results obtained by the closely relevant competitor Arabic-summarizers; and

---

\(^7\)SpaCy for Natural Language Processing https://spacy.io/usage/linguistic-features#sbd-component
TABLE 4. High-level types of sentence clauses and a comprehensive list containing Arabic conjunctions.

| First clause type | Conjunction | Second clause type |
|------------------|-------------|-------------------|
| Independent (main) clause | A: Independent clause | B: Dependent clause |
| Independent clause | C: Independent clause | D: Dependent clause |
| Dependent (or subordinate) clause | E: Independent clause | F: Dependent clause |

Conjunctions that:
1. link words, phrases, semi-sentences, or sentences, including relative pronouns.
2. indicate retraction and restriction words.
3. indicate contradiction.
4. indicate cause and effect.
5. indicate past and present.
6. indicate past and present or future.

| # | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| 1 | * | 17% | but | 1 | 2 |
| 2 | * | 17% | until | 1 | 4 |
| 3 | * | 17% | must | 1 | 4 |
| 4 | * | 17% | due to | 1 | 4 |
| 5 | * | 33% | from the | 1 | 4 |
| 6 | * | 33% | as that / as it | 1 | 1 |
| 7 | * | 33% | from / what | 1 | 1 |
| 8 | * | 33% | but / but it | 1 | 2 |
| 9 | * | 33% | because / due to | 1 | 4 |
| 10 | * | 33% | where / so that | 1 | 4 |
| 11 | * | 33% | in terms of | 1 | 4 |
| 12 | * | 33% | whereas | 1 | 4 |
| 13 | * | 33% | because of that | 1 | 4 |
| 14 | * | 33% | as that | 1 | 4 |
| 15 | * | 33% | on this | 1 | 4 |
| 16 | * | 33% | therefore / as a result of this | 1 | 4 |
| 17 | * | 50% | also | 1 | 1 |
| 18 | * | 50% | like that | 1 | 1 |
| 19 | * | 50% | who: single male/female | 1 | 1 |
| 20 | * | 50% | who: dual males/females | 1 | 1 |
| 21 | * | 50% | who: plural males/females | 1 | 1 |
| 22 | * | 50% | despite / although | 1 | 2 |
| 23 | * | 50% | despite of | 1 | 2 |
| 24 | * | 50% | even though | 1 | 2 |
| 25 | * | 50% | معي | 1 | 2 |

For the purpose of human evaluation, we created a separate dataset, available at². It comprises eight Arabic documents covering health, business, literature, and technology. Each article was written by a professional content writer and published on a reputable website. The statistics of this dataset is detailed in Table 8.

2) ROUGE METRICS
We utilized different metrics stemmed from the Recall-Oriented Understudy for Gisting Evaluation (ROUGE) [42] (i.e., ROUGE-1, ROUGE-2, ROUGE-L). More in detail, ROUGE gives various metrics for assessing textual-based models, popularly used in the evaluation of text translator and summarizer models. Its underlying idea is to identify the number of overlapping n-grams (or token subsequences) between the produced texts and the reference texts using the standard statistical measurements (precision, recall, and F-measure). ROUGE-1 and ROUGE-2 measure the overlap of unigram and bi-grams, respectively, whereas the overlapping in ROUGE-L depends on the longest identical subsequence between the produced texts and the reference texts.
TABLE 5. Statistical descriptions of the evaluation EASC dataset, determined after performing our preprocessing step.

| Category (# documents) | Art and Music | Politics | Education | Religion | Environment | Sci-Tech | Finance | Sport | Health | Tourisms |
|------------------------|--------------|----------|-----------|----------|-------------|---------|---------|-------|--------|----------|
| # Documents: 153       | 10           | 21       | 7         | 8        | 33          | 16      | 17      | 10    | 17     | 14       |

# Sentences: 2,440

| max per Docs. | min per Docs. | avg per Docs. | 16 |
|---------------|---------------|---------------|---|
| 3             | 93            | 117           | 49 |

| Words: 63,987 |
|---------------|
| min per Docs. | max per Docs. | distinct words | avg per Docs. | 1083 | 419 |

# Human-generated summaries: 5 per doc. (i.e., 5 × 153 = 765)

Here, higher ROUGE scores (i.e., higher overlapping n-grams) refer to better quality correlated to rational human judgments [2]. Given the number of overlapping n-grams between the generated summary $Y$ and a reference summary $doc_{Ref}$ (i.e., $nGrams_{overlapping} = |nGrams^Y \cap nGrams^{doc_{Ref}}|$), we compute ROUGE’s precision, recall, and F-measure using Equation 3, 4, and 5, respectively. To clarify more, ROUGE precision measures the relevant nGrams generated in $Y$. In contrast, ROUGE recall estimates how much nGrams of $Y$ are found in the reference summary $doc_{Ref}$. F-measure, however, gives a balancing score between the precision and recall scores. In this paper, we calculate these scores using a standard ROUGE script (https://pypi.org/project/rouge/).

\[
\text{Precision} = \frac{|nGrams_{overlapping}|}{|nGrams^Y|} \tag{3}
\]

\[
\text{Recall} = \frac{|nGrams_{overlapping}|}{|nGrams^{doc_{Ref}}|} \tag{4}
\]

\[
F\text{-measure} = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \tag{5}
\]

B. PERFORMANCE OF ArDBertSum WITH OPTIMIZING THE INITIAL DistilBERT SUMMARY

Table 6 shows the performance of ArDBertSum on the five (EASC) reference documents (a-e) in two runs: with and without using the proposed SCSAR. Through these obtained Rouge’s scores, while paying more attention to the average of Rouge-precisions (i.e., $p=0.39$, $p=0.32$, and $p=L=0.41$), we can observe that ArDBertSum performs well and presents notable competitive results. To give a clue on reading these scores, the average Rouge-precision scores of many non-Arabic state-of-the-art summarization approaches is relatively less than 0.21, see [31], [43], [44]. Besides that, an average of 0.33 was recently reported in [30], manipulating the BERT\texttt{large} model.

Furthermore, the practical feasibility of applying the second stage of ArDBertSum, depending on SCSAR, is confidently anticipated. Giving insight into the Rouge-F, see the average scores of using vs. without using SCSAR in Table 6, the obtained minor improvements indicate that the SCSAR managed not to negatively affect the quality of the final candidate summary (i.e., $Y$). Apart from such minor improvements, we imagine that SCSAR has a more significant effect, particularly with shorting long and complex sentences. Therefore, we have strived to scrutinize the effect of SCSAR through eliciting human judgments, discussed later in this section.

C. PERFORMANCE COMPARISON WITH THE RELATED EXTRACTIVE APPROACHES

To our knowledge, there exists a limited handful of Arabic text summarization approaches [9], [10], [13], [15], [45], [46] that are comparable to our ArDBertSum. In Table 7, we listed these extractive-based approaches as baselines and classified them into either heuristic or non-heuristic techniques. The reported results by these approaches, including ours, are based on the same EASC dataset. Here, we shed light on that the heuristic-based approaches (i.e., have derived knowledge,
TABLE 8. Statistics about the selected articles and ArDBertSum-generated summaries.

| Variable                      | Health | Business | Literature | Technology | Average |
|-------------------------------|--------|----------|------------|------------|---------|
|                              | Ar-1   | Ar-2     | Ar-1       | Ar-2       | Ar-1    | Ar-2    | M   | SD |
| Paragraphs                   | 14     | 11       | 12         | 13         | 8       | 7       | 11  | 14 | 11.25 | 2.60 |
| Lines                        | 51     | 49       | 29         | 56         | 44      | 45      | 27  | 49 | 43.75 | 10.40 |
| Words                        | 538    | 532      | 323        | 549        | 534     | 537     | 308 | 549| 483.75| 104.11|
| Characters (no space)        | 3003   | 2772     | 1481       | 2822       | 2708    | 2766    | 1505| 2694| 2468.87| 609.80|
| Characters (with space)      | 3528   | 3294     | 1792       | 3359       | 3235    | 3300    | 1803| 3229| 2942.5| 712.88|

TABLE 9. Aspects rated to evaluate the quality of automatic summaries.

| Aspect                        | Rating Statement                                      | Arabic Translation |
|-------------------------------|-------------------------------------------------------|---------------------|
| 1. Fluency                    | The summary is eloquently written                      | المفصل اعلان مكثر بالصرف |
| 2. No Content Duplication     | There is no duplicate content in the summary          | لا يوجد محتوى مكرر |
| 3. Errors                     | The summary contains errors                            | يحتوي المفصل على أخطاء |
| 4. Sufficient Details         | The summary provides sufficient details                | يقدم المفصل تفصيلا كافيا |
| 5. Overall rating             | In general, how would you rate the quality of the summary | يشكل عاما كيف تقيم جودة المفصل |

strategies, or any information from EASC dataset for optimizing their summarizer models) give a better precision score than non-heuristic approaches. Nevertheless, such heuristics might render these summarizers prone to inaccuracies and bias in evaluation. Heuristics in this context would include (1) training the summarizers using the EASC dataset, or taking advantage of the overall keyword-tokens across multiple original (EASC) documents [13], or examining this specific EASC dataset to extract useful heuristic (e.g., heuristic for the A* algorithm as in [10]).

Towards the generality, our ArDBertSum is entirely non-heuristic and depends on a fine-tuned version of a pre-trained language understanding model (i.e., DistilBERT). Considering the non-heuristic approaches, ArDBertSum gives the best performance (compared with mRMR [15] and LCEAS [46]) with minor improvement when using SCSAR. Moreover, the F-measures of all heuristic approaches, including ArDBertSum, look resembling, which is considered an outweighed trait in favor of ArDBertSum.

D. HUMAN EVALUATION

To solidify our assessment of the proposed ArDBertSum, we subjected its automatic summaries to human judgment. The evaluation was completed by users presented with the automatic text summaries and requested to assess several aspects of the text quality. The text summaries were administered as part of an online survey and shared with students for completion. The students were rewarded extra credits for their participation in the online survey.

The readers evaluated eight different text excerpts, covering four vital topics: health, business, literature, and technology. We varied the writing topics to cater to several writing styles and qualities. All text samples were written in the formal Arabic language and retrieved from professional websites, such as the Sky News Arabia and Marefa websites, to ensure correctness, quality, and objectivity. We refrained from selecting hostile or adverse articles (e.g., discussing political stances, wars, and disasters) that could instigate any negative or positive user feelings, which might influence their judgment of the quality of summaries. Therefore, the selected articles are all deemed to contain neutral content.

Table 8 presents a statistical overview of the dataset that we collated and used to test our ArDBertSum. As stated earlier, eight articles spanning across four categories were evaluated, with two different articles covering one topic. On average, each original article contains seven to 14 paragraphs with approximately 500 and 550 words, except for one article in the business category and one article in the technology category. All images were removed from the selected articles, and only text was displayed to the participants. Once we applied the ArDBertSum to the text articles, we obtained a one-paragraph long summary for each article. We set the SCSAR to a 75% reduction rate of the article size, resulting in 66 – 171 word summaries (i.e., keeping approximately 28% of the original length of the article). Seven summaries had more than 100 words, as shown in Table 8. The original text files and automatically-generated summaries of the articles are accessible at the accompanying GitHub repository for this paper². It is worth noting that the human judges were shown the summaries only and requested to rate their quality on several criteria.

VOLUME 9, 2021

135603
TABLE 10. Human judgment scores of automatic summarization of different topics; M = mean, SD = standard deviation; * = significant statistical difference.

| Variable                  | Health | Business | Literature | Technology | Overall | Statistical Significance |
|---------------------------|--------|----------|------------|------------|---------|--------------------------|
| Fluency*                  | 3.42   | 3.38     | 3.66       | 4.25       | 4.00    | 3.28                     |
| No Content Duplication*   | 3.58   | 3.16     | 4.19       | 4.22       | 4.01    | 3.08                     |
| Errors                    | 3.50   | 1.43     | 3.11       | 2.80       | 2.80    | 3.64                     |
| Sufficient Details        | 3.69   | 1.27     | 3.75       | 4.11       | 3.10    | 3.39                     |
| Overall rating            | 3.33   | 1.20     | 3.91       | 1.08       | 4.03    | 1.17                     |

In total, 18 users opted to take part in our human judgment of the summaries. 16 users were male (88.88%), and two users were females (11.11%). 16 users were students pursuing a bachelor’s degree, one high school, and one higher studies degree. 17 (94.44%) respondents are Arabic native speakers, while only one person has Arabic as a second language. However, when asked to rate their level of the Arabic language on a 5-point scale (where 1 = poor and 5 = excellent), the respondents indicated to have an excellent level (mean (M) = 4.22, standard deviation (SD) = 0.88).

To verify the effect of users’ interest on the evaluation scores, we gathered users’ ratings on the general topics that were judged. A one-way ANOVA test showed that our human judges had significantly different interests in the selected topics (F(3, 68) = 7.81, p = 0.0001). More specifically, post hoc T-tests revealed a higher interest in technology articles (m = 4.56, sd= 0.62) compared to business (m = 3.39, sd= 0.98), health (m = 3.22, sd=1.22), and literature articles (m=2.94, sd=1.35).

We adapted the evaluation methodology introduced in [47]. However, we felt the questionnaire could be enhanced to address some relevant characteristics which were missed. Therefore, the quality of our ArDBertSum-generated summaries was evaluated on five aspects. To this end, we have extended our questions to include five statements measuring the fluency, content duplication, errors, sufficient details, and overall rating of the summaries. These statements are listed, in English and Arabic, in Table 9. The complete survey, including summaries and questions, is available at (9).

A glance at Table 10, which summarises the average rating of human judges of the automatic summaries, indicates a non-significant variation in the scores based on the type of topics (p=0.08). Literature and business summaries received the highest quality scores (m=4.03 and m=3.91 respectively), while health and technology summaries received the lowest scores (m=3.33). On average, the summaries were rated close to good (m=3.65), which is quite acceptable. Across the five aspects that were assessed, content duplication received the highest rating (m=4.01), followed by the presence of sufficient detail (m=3.70) and text fluency (m=3.70). However, the evaluators thought the summaries contained language mistakes (m=3.26). It seems that the

9https://forms.gle/2ms2jZEQINjghtAC8

TABLE 11. User comments about the quality of the automatic summaries on the topics: Health, Business, Literature, and Technology.

| Topics | Comment | H | B | L | T | Total | Percent |
|--------|---------|---|---|---|---|-------|---------|
| + Positive | 3 | 2 | 5 | 1 | 11 | 22.45% |
| - Negative | 13 | 8 | 6 | 11 | 38 | 77.55% |
| Sub-Total | 16 | 10 | 11 | 12 | 49 | 100% |

perception that the summaries contain mistakes has impacted the perceived judgment of the quality of summaries. The topics varied significantly for the fluency and content duplication variables only (p<0.01). Technology summaries received the worst score for fluency (m= 3.28), while health summaries received the worst score for content duplication (m=3.58).

Moreover, when users were requested to write comments about the quality of each automatic summary, a total of 49 feedback were submitted, as shown in Table 11. Such qualitative data help us understand the rationale behind the assigned ratings. On average, each type of article invoked a similar number of comments, ranging between 11 to 16 comments. Indeed, the qualitative feedback mirror the quantitative rating; for instance, the health and technology summaries received 13 (81.25%) and 11 (91.66%) negative comments, respectively. 22.45% of the overall statements were positive, focusing mainly on the understandability of the summaries. However, the remaining 77.55% were negative, with the main concerns revolving around the vagueness of the summary, inappropriate use of punctuations, and incoherence of statements.

E. DISCUSSION OF FINDINGS AND THREATS TO VALIDITY

This work presents an Arabic summarizer approach, which combines the powers of the DistilBERT model and sentence-clause selection method (aka SCSAR) to create plausible extractive summaries. The findings of our experiment, using the EASC dataset, against seven extractive baselines (i.e., Gen-Summ, LSA-Summ, PSO, ArA* with SS, Multi-document, mRMR, and LCEAS) showed encouraging metric scores, particularly versus the non-heuristic methods. Moreover, the survey ratings of candidate summaries by human judges revealed acceptable perceptions of the summaries’ quality. However, these findings should be treated with caution until more experiments are carried out.
A thoughtful threat to validity that should be argued is why we have chosen DistilBERT over the other available pre-trained language-understanding models. While, recently, researchers have suggested different knowledge-transfer and language-understanding models that are competitors to DistilBERT for addressing the extractive ATS problem (i.e., depends on selecting key sentences only for producing a candidate summary), we are not principally interested in seeing the differences between their performance at the current state of our ongoing work. However, instead, we are interested in establishing our method upon a reliable and efficient model. To our thoughts, using other competitor models to DistilBERT would probably lead to getting the same conclusion of this paper. Moreover, evaluating extractive ATS solutions for Arabic texts, depending on an efficient/small knowledge transfer learning model such as DistilBERT, would be valuable to the Arabic NLP researchers. Certainly, we are not claiming that DistilBERT is the best choice for the ATS problem, nevertheless, examining this explicitly is considered a different research problem that is not within the focus of our work.

The advantages of our approach for automatic text summarization can be argued, particularly for the reduction of Arabic text, but it is entirely reliant on sentence selection. Although the ArDBertSum approach generally maintains the structure and syntax of the text, it does not warrant the correctness and coherence of the produced summary. In fact, this effect was observed in the qualitative comments of our human judges, where they indicated the discoordination of the sentences. Our reduction results are favorable; however, the benchmarking against other state-of-the-art approaches is not always fair and straightforward. This is because the comparison depends on a myriad of complex parameters, such as encoding heuristic, reduction rate, evaluation metrics, among others. Unifying the configuration of the experiments is no easy matter.

Although the proposed ArDBertSum approach demonstrated promising results, a few threats to validity are worth noting below. We acknowledge that our benchmarking against other approaches employed different experiment settings. Our model’s predictions were produced using the best parameters for our experiment. Similarly, we have assumed that the competitors’ results were obtained using other experimental settings. Our validation assessed the quality of the ArDBertSum-based automatic summarization using 1) a set of overlapping units derived using ROUGE and 2) a human evaluation study.

The ArDBertSum approach was applied to professionally written articles. Thus, its generability is limited to the formal Arabic language only. Informal Arabic summaries generated using our approach are likely to exhibit a different quality and, thereby, scores. Moreover, the quality of the generated summaries will highly correlate with the quality of the pre-trained BERT model used and its ability to understand the Arabic language. However, we anticipate that combining BERT and SCSAR will empower the generation of good-quality summaries. The evaluations conducted herewith used two small datasets, which restricts the generability of the proposed approach and begs for further studies in the future. The EASC dataset was adopted for the technical comparison, while a new dataset, covering four distinct domains, was created for human evaluation. This decision entails two benefits. First, we are making this new dataset available for other researchers to exploit in their studies. Second, for practical reasons, we used this small dataset to guide the human evaluation.

There is a mixed argument in the literature regarding whether involving students in experimental research through incentives provides meaningful results. Omori and Feldhaus [48] found no link between using incentives and the quality of collected data. Furthermore, the readers’ interest does not seem to influence their judgment of the quality of summaries. For instance, although the human judges were highly interested in technology, they rated the technology summaries the worst. Finally, we would like to highlight that although the human judges were Arabic native speakers, they were not pretested to ensure their proficiency in the Arabic language. In future studies, Arabic language experts will be recruited to create gold standard summaries for benchmarking purposes and evaluate the summaries for an accurate estimation of the quality of the summaries. Moreover, we plan to conduct additional comparative studies against prominent Arabic language text summarization techniques to assert the superiority of our approach.

V. CONCLUSION
Automatic Text Summarization is deemed as one of the most complex NLP applications, particularly for the Arabic language that has not been intelligently developed like the other Indo-European languages. Towards producing a summarizer for text written in Arabic, relying on a pre-trained Language Understanding model (LUM), this paper has examined the ability of a fine-tuned version of DistilBERT in addressing the Arabic ATS concluded with offering a summarizer (ArDBertSum). To shorten long or complex sentences using our ArDBertSum, we have attempted to segment them into possibly independent clauses using the proposed SCSAR.

Furthermore, we have carried out several experiments to assess the performance of ArDBertSum (with and without using SCSAR) against the related competitor Arabic summarizers. The experiments were conducted on two datasets (one of which is a well-known EASC corpus) utilizing ROUGE metrics. While ensuring the quality of candidate summaries produced by most of the ATS tools available today is a state of uncertainty, we have also conducted a specific human-based evaluation to verify the obtained performance measures. The human judges rated eight automatically-generated summaries, covering various domains, on five main criteria. Although a 75% text reduction rate was applied to the original articles, the judges deemed the produced summaries as acceptable. The main observations, however, focused on the coherence of the sentences and use of punctuations. The results of our experiments illustrated that ArDBertSum yields the best performance, compared with non-heuristic Arabic
summarizers, in producing an acceptable quality of candidate summaries.

A notable limitation to mention (i.e., besides the uncertainty problem in evaluating the ATS systems) is that the dataset size (EASC) may not be comprehensively enough to evaluate the suggested ATS solutions. Therefore, towards the generality, we imagine a more refinement and extension of this ongoing work in three directions. The first direction is to evaluate our ArDBertSum on other datasets. Since there is a noticeable lack of available datasets for the Arabic ATS, we plan to create a specific benchmarking dataset, focusing on the formal Arabic texts. The second direction would be to study the evaluation metric for ATS by probably combining statistical metrics (e.g., ROUGE) along with human estimation. The third natural extension to this work would be to combine the abstractive summarization technique as a feature in our ATS solution. Rather than confining our approach to DistilBERT only where other significant competitors’ models have emerged (such as Longformer, GPT-2, RoBERTa, or XLMs), this third extension should incorporate a practical comparison among the state-of-the-art pre-trained NLU models in order to base our approach on the most appropriate version.

REFERENCES

[1] 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.
[2] L. M. Al Qassem, D. Wang, Z. Al Mahmoud, H. Barada, A. Al-Rubaie, and N. I. Almoosa, “Automatic Arabic summarization: A survey of methodologies and systems,” Proc. Comput. Sci., vol. 117, pp. 10–18, Jan. 2017.
[3] H. P. Luhn, “The automatic creation of literature abstracts,” IBM J. Res. Develop., vol. 2, no. 2, pp. 159–165, Apr. 1958.
[4] M. T. B. Othman, M. A. Al-Hagery, and Y. M. E. Hashemi, “Arabic text summarization process: Verbs roots and conjugation automation,” IEEE Access, vol. 8, pp. 103913–103923, 2020.
[5] R. Alqaqi, W. Ghanem, and A. Qarouch, “Extractive multi-document Arabic text summarization using evolutionary multi-objective optimization with K-medoid clustering,” IEEE Access, vol. 8, pp. 228206–228224, 2020.
[6] A. M. Azmi and S. Al-Thanyan, “A text summarizer for Arabic,” Comput. Speech Lang., vol. 26, no. 4, pp. 260–273, Aug. 2012.
[7] R. Badry and I. F. Mosawad, “A semantic text summarization model for Arabic topic-oriented,” in Proc. 4th Int. Conf. Adv. Mach. Learn. Technol. Appl. (AMLTIA), vol. 921, Cairo, Egypt: SRGE, Mar. 2019, pp. 518–528.
[8] N. Alami, M. El Mallah, H. Amakdouf, and H. Qidiaa, “Hybrid method for text summarization based on statistical and semantic treatment,” Multimedia Tools Appl., vol. 80, no. 13, pp. 3–34, 2021.
[9] 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.
[10] B. Bahloul, H. Aliane, and M. Benmohammed, “ARA*summarizer: An Arabic text summarization system based on subtopic segmentation and using an A* algorithm for reduction,” Expert Syst., vol. 37, no. 2, Apr. 2020, Art. no. e12476.
[11] 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.
[12] M. Alhoshan and N. Altwaijry, “AUSS: An Arabic query-based update-summation system,” J. King Saud Univ. Comput. Inf. Sci., Dec. 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1319157820305565?via%3Dihub, doi: 10.1016/j.jsuci.2020.11.027.
[13] S. Abdulateef, N. A. Khan, B. Chen, and X. Shang, “Multidocument Arabic text summarization based on clustering and Word2Vec to reduce redundancy,” Information, vol. 11, no. 2, p. 59, Jan. 2020.
ABDULLAH ALSHANQITI received the B.Sc. degree in computer science from Taibah University, Madinah, Saudi Arabia, and the M.Sc. and Ph.D. degrees from the University of Leicester, Leicester, U.K. He joined the Faculty of Computer Science and Information Technology (FCIS), in 2012, as a Lecturer. He was an Assistant Professor in smart systems and software reverse engineering, in 2018. He is currently the Vice Dean of FCIS, Islamic University of Madinah, recognized for his work on machine learning, software reverse engineering based on dynamic analysis, model/graph transformations using intelligent learning, and inference approaches. His research interests include research cooperation in different cutting-edge disciplines, including quantum machine learning, hybrid AI approaches that focus on solving NLP, computer vision challenges, and interpretability of deep learning models using graph transformations rules.

ABDULLAH NAMOUN (Member, IEEE) received the bachelor’s degree in computer science from The University of Manchester, U.K., in 2004 and 2009, respectively. He is currently an Associate Professor of intelligent interactive systems and the Head of the Information Systems Department, Faculty of Computer and Information Systems, Islamic University of Madinah. He has authored more than 50 publications in research areas spanning intelligent systems, human–computer interaction, software engineering, and technology acceptance and adoption. He has extensive experience in leading complex research projects (worth more than 21 million Euros) with several distinguished SMEs, such as SAP, BT, and ATOS. He has investigated user needs and interaction with modern interactive technologies, design of composite software services, and methods for testing the usability and acceptance of human-interfaces. His research interests include integrating state of the art artificial intelligence approaches in the design and development of interactive systems.

AISHA MOUSA MASHRAQI received the bachelor’s degree in computer science from Princess Nourah Bint Abdulrahman University, Riyadh, Saudi Arabia, in 2008, and the master’s and Ph.D. degrees in computer science from the University of Leicester, Leicester, U.K., in 2013 and 2018, respectively. In 2011, she joined the Department of Computer Science, College of Computer Science and Information Systems, Najran University, as a Lecturer, and became an Assistant Professor, in 2018.

ABDUL REHMAN GILAL received the Ph.D. degree in information technology from Universiti Teknologi PETRONAS (UTP), Malaysia. He is currently a Faculty Member with the Department of Computer Science, Sukkur IBA University, Pakistan. His research interest includes software project management for finding the effective methods of composing software development teams. Based on his research publication track record, he has contributed in the areas of human factor in software development, complex networks, databases and data mining, programming, and cloud computing.

SAMI SAAD ALBOUQ received the B.S., M.S., and Ph.D. degrees in computer science, in 2009, 2011, and 2018, respectively. He is currently an Assistant Professor with the College of Computer Science and the Dean of Information Technology Deanship with the Islamic University of Madinah, Madinah, Saudi Arabia. His research interests include system security engineering, complex information technology systems, and quantum key distribution systems.