Indonesian Journal of Computer Science
ISSN 2549-7286 (*online*)
Jln. Khatib Sulaiman Dalam No. 1, Padang, Indonesia
Website: ijcs.stmikindonesia.ac.id | E-mail: ijcs@stmikindonesia.ac.id

LEMMA-ROUGE: An Evaluation Metric for Arabic Abstractive Text Summarization

Amal M. Al-Numai, Aqil M. Azmi
Email amal.alnumai@gmail.com, aqil@ksu.edu.sa
Department of Computer Science, College of Computer and Information Sciences, King Saud University, Riyadh 11543, Saudi Arabia

Article Information

Submitted : 17 Apr 2023
Reviewed : 25 Apr 2023
Accepted : 27 Apr 2023

Keywords

Evaluation, Abstractive Summary, Arabic, Summarization, ROUGE, Evaluation Metric

Abstract

High morphological languages are characterized by complex inflections and derivations, which can present challenges for natural language processing tasks such as summarization. Abstractive text summarization aims to generate a summary by understanding the meaning of the text, rather than solely relying on the words used in the original source. However, few works address the generation of abstractive summaries due to its complexity. One of the challenges is the absence of a reliable metric to evaluate the performance of abstractive summaries. This paper proposes a lemma-based ROUGE metric and investigates the effectiveness of normalization forms in the similarity matching of the ROUGE metric for evaluating abstractive text summarization systems. We use Arabic as a case study and compare results involving different word forms: as is, stem-based, and lemma-based. The results show that the lemma-based form achieves higher ROUGE scores than the other forms. The findings emphasize the impact of morphological complexity on the performance of abstractive text summarization systems.

A. Introduction

Abstractive Text Summarization (ATS) is a challenging natural language processing (NLP) task that involves generating a concise and coherent summary of a given document while preserving its essential meaning. Unlike extractive summarization, which selects and concatenates important sentences from the source text, abstractive summarization paraphrases sentences or generates new ones that may not be present in the original text. Table 1 shows a sample text along with both types of summaries. In this example, both summaries capture the main point of the original text. However, the extractive summary takes the wording directly from the original text and condenses it, while the abstractive summary rephrases it using simpler language and omits some details.

Table 1. Example text and its sample extractive and abstractive summaries

Original text	Extractive summary	Abstractive summary
Tom and Jerry went by bicycle to listen to a lecture on campus. While in class, Tom got a call and headed back home.	Tom and Jerry listen to lecture on campus. Tom headed home.	Tom headed home after listening to a lecture with Jerry.

Abstractive summaries offer several benefits over extractive ones, particularly in their ability to convey the essence of the original text. This feature is particularly useful for addressing, for example, text simplification, which is a difficult task in NLP. By utilizing appropriate language based on the target reader's level, abstractive summaries can help overcome this challenge.

Overall, abstractive summaries are typically more informative, concise, and versatile than extractive summaries, but they can also be more challenging to create and may not always be as faithful to the original text. This raises a question. How good is the generated abstractive summary?

In this work, we introduce a variant version of a metric that is being used to evaluate the summaries. We call it LEMMA-ROUGE, an adaptation of the standard ROUGE evaluation metric aimed at overcoming the limitations arising from the unique features of the Arabic language.

B. Evaluating Summaries

The evaluation's objective is to assess a model's effectiveness in determining how well it performs in generating summaries that capture the most important information from the source text. Evaluation provides feedback on what aspects of a summary need improvement, such as coherence or relevance. Evaluation allows for comparison between different abstractive text summarization models, which can help identify which model performs better than others [1]. It helps track progress in text summarization research over time as new techniques and approaches are developed.

There are two categories of summary evaluation measures: intrinsic and extrinsic. Intrinsic measures evaluate the quality and content of the summary. Quality is measured by how easy it is to read and is typically evaluated manually by human judges. Content evaluation varies depending on what is being summarized. Sentence extracts are often evaluated using co-selection, while human abstracts

achieve better results with content-based measures. Extrinsic measures, on the other hand, use task-based methods to evaluate the performance of a summary for a specific task, such as information retrieval. For our purpose, which pertains to abstractive text summarization, we are mostly interested in intrinsic measures.

In co-selection, the main evaluation metrics are the well-known measures: precision, recall, and F_1-score. We will focus on the content-based measure. Consider the following two summaries describing the scenario where a child must choose between a chocolate or vanilla ice cream: (a) The child picked chocolate, and (b) He didn't choose vanilla. Both summaries convey the same information, but the wording is completely different. The F_1-scores cannot capture this semantic difference, which means a metric for evaluating the semantic similarity between summary vectors is necessary.

Abstractive text summarization can be evaluated manually and automatically. Manual evaluation involves human summarizers reading the generated summary and evaluating its quality based on criteria such as coherence, fluency, informativeness, relevance etc. [2, 3]. Human evaluators may have different opinions on what constitutes a good summary, leading to subjective evaluations that may not be consistent across different annotators. They may have biases towards certain types of summaries or topics, leading to a lack of diversity in the evaluated summaries. Therefore, manual evaluation is limited by the ability of human evaluators to identify all aspects of summary quality which can lead to incomplete evaluations that do not fully capture system performance. In addition, it is time-consuming and expensive, especially for large datasets as it is not scalable to large datasets or real-time applications where summaries need to be generated quickly.

Automated evaluation metrics have been developed to overcome these limitations. BLEU (BiLingual Evaluation Understudy) [4] was first used to evaluate machine translation task. BLEU counts how many *n*-grams in the system model output appear in the reference translations. Wazery et al. [5] used BLEU to evaluate their summarizer. BLEU considers the surface-level similarity between the system summary and the reference summaries, which is not considered a good metric for evaluating abstractive summarization task. METEOR (Metric for Evaluation of Translation with Explicit ORdering) [6] is another machine translation metric designed to overcome the limitations of BLEU. It incorporates stemming if the word's form of the system model and reference translations differed. It was used to evaluate the summarization task [7]. BERTScore [8] is yet another metric that computes a similarity score between the system sentence and the reference sentence based on pre-trained BERT contextual embeddings. AraBART model [9] evaluated using BERTScore.

Heretofore, ROUGE (Recall-Oriented Understudy for Gisting Evaluation) [10] is the most used metric for evaluating abstractive text summarization tasks and for comparing summarization models. It measures the similarity between a system generated summary and one or more reference summaries based on *n*-gram overlap. It calculates the overlapping using various measures such as precision, recall, and F_1 score. The most used ROUGE measures are ROUGE-1, ROUGE-2, and ROUGE-L. ROUGE-1 measures the overlap of unigrams (single words), while ROUGE-2 measures the overlap of bigrams (pairs of adjacent words). ROUGE-L is based on the longest common subsequence (LCS) between the system generated

summary and reference summaries. ROUGE is a recall-based metric that focuses on how much information from the reference summaries is captured in the system generated summary. It has been used to evaluate various abstractive summarization models [3, 5, 9, 11–13], covering different techniques such as deep learning and graph-based models [14, 15].

ROUGE is valuable for evaluating summarization systems that aim to capture as much relevant information as possible in a shorter form. However, it has limitations, especially when used with high morphological languages. It relies on exact word matching and does not consider the inflectional and derivational morphology of these languages. For example, in Arabic, a word can have multiple forms depending on its grammatical function in a sentence. This can lead to mismatches between the system and reference summaries, affecting the ROUGE score. Suleiman et al. [13] reshaped ROUGE by ignoring word ordering matching, replacing the words in the system summary with their stem form, and using cosine similarity to measure the semantic similarity of a word based on a pre-trained model. Nevertheless, using stem-based matching will ignore the semantic meaning of words.

C. Proposed Metric LEMMA-ROUGE

We propose a modified version of the ROUGE metric, called LEMMA-ROUGE, which is designed to account for the linguistic characteristics of the Arabic language. The lemmatization process involves reducing words to their lemma form, which can aid in capturing their underlying meanings and minimize variations in word forms. This approach is frequently utilized in various NLP tasks to standardize distinct surface-level words during text processing for comparisons and matching purposes.

Our proposed method to overcome the problem of exact word matching in ROUGE is by modifying the ROUGE metric to consider the lemma form instead of the exact word. ROUGE includes multiple measures such as ROUGE-N, ROUGE-L, ROUGE-S, and ROUGE-SU. ROUGE-N is defined by the Equation,

$$\text{ROUGE-N} = \frac{\sum_{S \in RS} \sum_{n\text{-gram} \in S} \text{count}_{\text{match}}(n\text{-gram})}{\sum_{S \in RS} \sum_{n\text{-gram} \in S} \text{count}(n\text{-gram})},$$

where *S* is the reference summary, *RS* is the set of reference summaries, *n*-gram is a subsequence of *n* words from a given text, N (in ROUGE-N) is the length of the *n*-gram, $\text{count}_{\text{match}}(n\text{-gram})$ is the maximum number of matched *n*-gram words between the reference summary and the system summary, and count(*n*-gram) is the total number of *n*-gram words in the reference summary.

ROUGE-L is the longest common subsequence between the reference and system summaries. ROUGE-S (Skip-Bigram Co-Occurrence) measures the overlap of skip-bigrams between the reference and system summaries. ROUGE-SU is similar to ROUGE-S with the addition of counting unigrams.

Our proposal is the LEMMA-ROUGE metric, which involves converting both the words in the reference and system summaries to their respective lemma forms

as a unified surface-level representation. We then apply the standard ROUGE measures to the reference and system summaries. The modified version of the ROUGE-N is calculated as follows:

$$\text{LEMMA\_ROUGE-N} = \frac{\sum_{S \in RS} \sum_{\text{lemma\_}n\text{-gram} \in S} \text{count}_{\text{match}}(\text{lemma\_}n\text{-gram})}{\sum_{S \in RS} \sum_{\text{lemma\_}n\text{-gram} \in S} \text{count}(\text{lemma\_}n\text{-gram})},$$

where lemma_*n*-gram is a subsequence of *n* words' lemma from a given text.

D. Results and Discussion

To measure the effectiveness of the proposed metric, we evaluate Arabic abstractive summarization models using ROUGE and LEMMA_ROUGE metrics. For a brief look at single-document abstractive summarizers developed during the last decade, see [16].

We aim to evaluate the proposed metric on document-level Arabic abstractive summarizers. Therefore, two Arabic abstractive text summarizers were considered for measuring the effectiveness of LEMMA_ROUGE metric; our proposed abstractive Arabic text summarizer based on Ant Colony System (AASAC) [17] and Azmi et al. abstractive summarizer [18], denoted ANSum. AASAC used the Ant Colony System to construct a short path solution followed by a text generation model that generates a summary. ANSum is an abstractive summarizer built on top of an extractive summarizer [19, 20], which in turn is based on Rhetorical Structure Theory (RST). After generating an extractive summary, ANSum post-processes it by shortening the output's sentences. This is achieved by removing specific words such as position names, days, and sub-sentences to create a more concise and coherent abstractive summary.

Due to the lack of a gold-standard dataset for Arabic single-document abstractive summaries, we have utilized the dataset collected by [18], which we have named ANDataset. However, for our experiments, we will only use a subset of this dataset, consisting of 104 documents with 30% and 50% summary sizes generated by the system. These documents were collected from Arabic newspapers and covered various topics such as general health, sports, politics, business, and religion, with an average length of 239 words.

Lemmatization was applied to the reference summaries of ANDataset, and the system generated summaries of ANSum and AASAC summarizers using the Stanza toolkit [21] and Farasa Lemmatization [22]. Additionally, to measure the effect of lemmatization over stemming, we used Tashaphyne [23], ISRIStemmer [24], and ARLSTem2 [25] stemmers to construct stem-form system and reference summaries. Following that, the ROUGE-2.0 toolkit [26] was used to evaluate ANSum [18] and AASAC summarizers [17].

Results are reported for different types of ROUGE; ROUGE-1, ROUGE-2, ROUGE-3, ROUGE-4, ROUGE-L, ROUGE-S4, and ROUGE-SU4. Recall, precision, and F-measure were calculated for each variant. The stem-based ROUGE measurement is denoted as STEM_ROUGE and lemma-based ROUGE is expressed as LEMMA_ROUGE.

Tables 2-3 show the results of different ROUGE metrics for the summaries that were generated via AASAC with 30% and 50% summary length, respectively. The best results are boldfaced. It is worth noting that stemming and lemmatization give better results than vanilla ROUGE scores on the original system and reference text. Moreover, our LEMMA_ROUGE metric outperforms ROUGE stem-based metric on all ROUGE types and scales with Stanza lemmatizer.

Table 2. Comparison between original ROUGE, STEM_ROUGE, and LEMMA_ROUGE values for AASAC summarizer [17] with 30% summary. Best results are in bold.

	Recall					
		STEM_ROUGE			**LEMMA_ROUGE**	
ROUGE-Type	**Original ROUGE**	**Tashaphyne**	**ISRIStemmer**	**ARLSTem2**	**Farasa**	**Stanza**
ROUGE-1	0.2615	0.3062	0.3501	0.3630	0.3644	**0.4347**
ROUGE-2	0.0937	0.1129	0.1415	0.1513	0.1434	**0.1980**
ROUGE-3	0.0385	0.0469	0.0631	0.0702	0.0632	**0.1020**
ROUGE-4	0.0161	0.0201	0.0297	0.0341	0.0298	**0.0569**
ROUGE-L	0.2278	0.2640	0.2928	0.3021	0.3345	**0.3836**
ROUGE-S4	0.0662	0.0861	0.1104	0.1175	0.1162	**0.1735**
ROUGE-SU4	0.1106	0.1361	0.1648	0.1732	0.1724	**0.2313**
	Precision					
		STEM_ROUGE			**LEMMA_ROUGE**	
ROUGE-Type	**Original ROUGE**	**Tashaphyne**	**ISRIStemmer**	**ARLSTem2**	**Farasa**	**Stanza**
ROUGE-1	0.3272	0.3816	0.4351	0.4501	0.4512	**0.6241**
ROUGE-2	0.1120	0.1345	0.1682	0.1790	0.1700	**0.2782**
ROUGE-3	0.0437	0.0530	0.0715	0.0789	0.0713	**0.1400**
ROUGE-4	0.0172	0.0214	0.0319	0.0362	0.0317	**0.0760**
ROUGE-L	0.3272	0.3763	0.4122	0.4258	0.4510	**0.5391**
ROUGE-S4	0.0730	0.0946	0.1210	0.1283	0.1274	**0.2367**
ROUGE-SU4	0.1251	0.1535	0.1855	0.1944	0.1938	**0.3190**
	F-Measure					
		STEM_ROUGE			**LEMMA_ROUGE**	
ROUGE-Type	**Original ROUGE**	**Tashaphyne**	**ISRIStemmer**	**ARLSTem2**	**Farasa**	**Stanza**
ROUGE-1	0.2889	0.3376	0.3856	0.3994	0.4003	**0.5099**
ROUGE-2	0.1014	0.1220	0.1528	0.1630	0.1545	**0.2301**
ROUGE-3	0.0407	0.0495	0.0667	0.0739	0.0665	**0.1174**
ROUGE-4	0.0165	0.0206	0.0306	0.0349	0.0305	**0.0647**
ROUGE-L	0.2671	0.3083	0.3402	0.3513	0.3817	**0.4460**
ROUGE-S4	0.0690	0.0896	0.1148	0.1220	0.1207	**0.1991**
ROUGE-SU4	0.1167	0.1434	0.1735	0.1821	0.1812	**0.2666**

Table 3. Comparison between original ROUGE, STEM_ROUGE, and LEMMA_ROUGE values for AASAC summarizer with 50% summary length

	Recall					
		STEM_ROUGE			**LEMMA_ROUGE**	
ROUGE-Type	**Original ROUGE**	**Tashaphyne**	**ISRIStemmer**	**ARLSTem2**	**Farasa**	**Stanza**
ROUGE-1	0.3682	0.4200	0.4657	0.4854	0.4879	**0.5573**
ROUGE-2	0.1709	0.2017	0.2471	0.2671	0.2502	**0.3410**
ROUGE-3	0.0865	0.1031	0.1387	0.1560	0.1371	**0.2241**
ROUGE-4	0.0445	0.0536	0.0791	0.0923	0.0770	**0.1513**
ROUGE-L	0.3354	0.3866	0.4118	0.4288	0.4812	**0.5353**
ROUGE-S4	0.1320	0.1686	0.2053	0.2225	0.2192	**0.3027**
ROUGE-SU4	0.1840	0.2238	0.2624	0.2803	0.2781	**0.3576**
	Precision					
		STEM_ROUGE			**LEMMA_ROUGE**	
ROUGE-Type	**Original ROUGE**	**Tashaphyne**	**ISRIStemmer**	**ARLSTem2**	**Farasa**	**Stanza**
ROUGE-1	0.4150	0.4733	0.5233	0.5460	0.5521	**0.7174**
ROUGE-2	0.1860	0.2193	0.2682	0.2902	0.2736	**0.4339**
ROUGE-3	0.0904	0.1075	0.1447	0.1631	0.1442	**0.2817**
ROUGE-4	0.0443	0.0533	0.0790	0.0924	0.0774	**0.1875**
ROUGE-L	0.4516	0.5137	0.5431	0.5686	0.6020	**0.6923**
ROUGE-S4	0.1347	0.1719	0.2091	0.2272	0.2252	**0.3781**
ROUGE-SU4	0.1917	0.2332	0.2730	0.2920	0.2917	**0.4494**
	F-Measure					
		STEM_ROUGE			**LEMMA_ROUGE**	
ROUGE-Type	**Original ROUGE**	**Tashaphyne**	**ISRIStemmer**	**ARLSTem2**	**Farasa**	**Stanza**
ROUGE-1	0.3888	0.4433	0.4910	0.5120	0.5158	**0.6250**
ROUGE-2	0.1776	0.2095	0.2564	0.2772	0.2604	**0.3805**
ROUGE-3	0.0882	0.1049	0.1412	0.1590	0.1400	**0.2487**
ROUGE-4	0.0443	0.0533	0.0789	0.0921	0.0769	**0.1669**
ROUGE-L	0.3812	0.4367	0.4636	0.4840	0.5298	**0.5987**
ROUGE-S4	0.1330	0.1697	0.2065	0.2241	0.2213	**0.3349**
ROUGE-SU4	0.1871	0.2276	0.2667	0.2850	0.2835	**0.3968**

Likewise, Tables 4-5 show the results of different ROUGE metrics for the summaries generated via ANSum with 30% and 50% summary length, respectively. It is noticeable that stemming and lemmatization give better results than computing ROUGE scores on the original system and reference text. Similarly, ROUGE lemma-based metric outperforms the ROUGE stem-based metric on all ROUGE types and scales.

Table 4. Comparison between original ROUGE, STEM_ROUGE, and LEMMA_ROUGE values for ANSum summarizer [18] with 30% summary length

	Recall					
		STEM_ROUGE			**LEMMA_ROUGE**	
ROUGE-Type	**Original ROUGE**	**Tashaphyne**	**ISRIStemmer**	**ARLSTem2**	**Farasa**	**Stanza**
ROUGE-1	0.2163	0.2254	0.2305	0.2377	0.2663	**0.2795**
ROUGE-2	0.1496	0.1457	0.1470	0.1596	0.1799	**0.1852**
ROUGE-3	0.1174	0.1105	0.1123	0.1260	0.1415	**0.1489**
ROUGE-4	0.0933	0.0864	0.0879	0.1009	0.1138	**0.1253**
ROUGE-L	0.2323	0.2453	0.2504	0.2562	0.3030	**0.3252**
ROUGE-S4	0.1265	0.1238	0.1258	0.1347	0.1597	**0.1682**
ROUGE-SU4	0.1469	0.1469	0.1496	0.1581	0.1838	**0.1928**
	Precision					
		STEM_ROUGE			**LEMMA_ROUGE**	
ROUGE-Type	**Original ROUGE**	**Tashaphyne**	**ISRIStemmer**	**ARLSTem2**	**Farasa**	**Stanza**
ROUGE-1	0.5311	0.5511	0.5645	0.5781	0.6387	**0.6987**
ROUGE-2	0.3561	0.3444	0.3490	0.3730	0.4178	**0.4616**
ROUGE-3	0.2687	0.2522	0.2575	0.2829	0.3156	**0.3687**
ROUGE-4	0.2031	0.1889	0.1928	0.2160	0.2426	**0.3064**
ROUGE-L	0.5144	0.5257	0.5302	0.5440	0.6022	**0.6576**
ROUGE-S4	0.2815	0.2749	0.2797	0.2966	0.3496	**0.4144**
ROUGE-SU4	0.3347	0.3338	0.3404	0.3566	0.4110	**0.4775**
	F-Measure					
		STEM_ROUGE			**LEMMA_ROUGE**	
ROUGE-Type	**Original ROUGE**	**Tashaphyne**	**ISRIStemmer**	**ARLSTem2**	**Farasa**	**Stanza**
ROUGE-1	0.2958	0.3080	0.3152	0.3244	0.3620	**0.3838**
ROUGE-2	0.2023	0.1965	0.1986	0.2147	0.2416	**0.2537**
ROUGE-3	0.1564	0.1471	0.1498	0.1670	0.1872	**0.2034**
ROUGE-4	0.1219	0.1131	0.1152	0.1313	0.1478	**0.1701**
ROUGE-L	0.3098	0.3246	0.3302	0.3380	0.3915	**0.4207**
ROUGE-S4	0.1670	0.1633	0.1661	0.1774	0.2100	**0.2293**
ROUGE-SU4	0.1956	0.1954	0.1992	0.2100	0.2436	**0.2633**

Table 5. Comparison between original ROUGE, STEM_ROUGE, and LEMMA_ROUGE values for ANSum summarizer with 50% summary length

	Recall					
		STEM_ROUGE			**LEMMA_ROUGE**	
ROUGE-Type	**Original ROUGE**	**Tashaphyne**	**ISRIStemmer**	**ARLSTem2**	**Farasa**	**Stanza**
ROUGE-1	0.3547	0.3593	0.3625	0.3756	0.3995	**0.4109**
ROUGE-2	0.2834	0.2761	0.2767	0.2996	0.3155	**0.3240**
ROUGE-3	0.2445	0.2357	0.2361	0.2656	0.2764	**0.2887**
ROUGE-4	0.2130	0.2042	0.2043	0.2376	0.2467	**0.2625**
ROUGE-L	0.3798	0.3894	0.3941	0.4041	0.4464	**0.4704**
ROUGE-S4	0.2585	0.2536	0.2541	0.2744	0.2962	**0.3063**
ROUGE-SU4	0.2796	0.2768	0.2779	0.2967	0.3189	**0.3289**
	Precision					
		STEM_ROUGE			**LEMMA_ROUGE**	
ROUGE-Type	**Original ROUGE**	**Tashaphyne**	**ISRIStemmer**	**ARLSTem2**	**Farasa**	**Stanza**
ROUGE-1	0.6961	0.7057	0.7114	0.7371	0.7756	**0.8221**
ROUGE-2	0.5398	0.5265	0.5277	0.5719	0.5965	**0.6532**
ROUGE-3	0.4513	0.4354	0.4364	0.4926	0.5079	**0.5865**
ROUGE-4	0.3804	0.3649	0.3654	0.4274	0.4392	**0.5364**
ROUGE-L	0.6878	0.6873	0.6882	0.7117	0.7458	**0.7825**
ROUGE-S4	0.4702	0.4620	0.4630	0.5012	0.5355	**0.6228**
ROUGE-SU4	0.5168	0.5122	0.5142	0.5499	0.5849	**0.6663**
	F-Measure					
		STEM_ROUGE			**LEMMA_ROUGE**	
ROUGE-Type	**Original ROUGE**	**Tashaphyne**	**ISRIStemmer**	**ARLSTem2**	**Farasa**	**Stanza**
ROUGE-1	0.4622	0.4684	0.4724	0.4894	0.5185	**0.5383**
ROUGE-2	0.3654	0.3562	0.3570	0.3865	0.4055	**0.4255**
ROUGE-3	0.3115	0.3005	0.3010	0.3387	0.3513	**0.3797**
ROUGE-4	0.2679	0.2570	0.2572	0.2993	0.3096	**0.3456**
ROUGE-L	0.4824	0.4907	0.4947	0.5083	0.5516	**0.5803**
ROUGE-S4	0.3274	0.3214	0.3220	0.3478	0.3739	**0.4028**
ROUGE-SU4	0.3563	0.3529	0.3543	0.3783	0.4049	**0.4321**

The results presented here indicate that STEM_ROUGE results are consistent with those reported in [13], which also showed that stemming improves ROUGE scores. Additionally, the ROUGE lemma-based metric provides a better measure of the informativeness of the system summary compared to other ROUGE surface-based metrics. This is particularly clear for summaries that are 30% or more of the original text, such as 50%. The reduction in STEM_ROUGE scores relative to

LEMMA_ROUGE may be due to the extensive removal of affixes in the stemming process, which can result in the formation of non-real words in some cases. These errors can be corrected using various techniques discussed in [27].

E. Conclusion

Evaluation is a crucial aspect of the text summarization task which is used to determine the quality of a generated summary. ROUGE is a widely used metric for evaluating different natural language processing tasks, including abstractive text summarization, although it has some limitations, especially for high morphological languages. It measures the lexical similarity between generated summaries and reference summaries. In this paper, we incorporated lemmatization to overcome this limitation. The results confirmed that lemmatization improves ROUGE scores for abstractive Arabic text summarization tasks. While manual evaluation remains an important tool for evaluating abstractive text summarization systems' performance in some contexts, such as small-scale text, automated metrics such as ROUGE are widely used due to their speed and scalability in large-scale evaluations.

F. References

[1] H. Y. Koh, J. Ju, M. Liu, and S. Pan, "An Empirical Survey on Long Document Summarization: Datasets, Models, and Metrics," *ACM Comput Surv*, vol. 55, no. 8, pp. 1–35, 2022.

[2] R. Paulus, C. Xiong, and R. Socher, "A Deep Reinforced Model for Abstractive Summarization," in *International Conference on Learning Representations*, 2018 [Online]. Available: https://openreview.net/forum?id=HkAClQgA-

[3] M. Kahla, Z. G. Yang, and A. Novák, "Cross-Lingual Fine-tuning for Abstractive Arabic Text Summarization," in *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021)*, 2021, pp. 655–663.

[4] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: a method for automatic evaluation of machine translation," in *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, 2002, pp. 311–318.

[5] Y. M. Wazery, M. E. Saleh, A. Alharbi, and A. A. Ali, "Abstractive Arabic Text Summarization Based on Deep Learning," *Comput Intell Neurosci*, vol. 2022, pp. 1–14, 2022.

[6] S. Banerjee and A. Lavie, "METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments," in *Proceedings of the ACL workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, 2005, pp. 65–72.

[7] A. See, P. J. Liu, and C. D. Manning, "Get To The Point: Summarization with Pointer-Generator Networks," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2017, vol. 1, pp. 1073–1083.

[8] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi, "BERTScore: Evaluating Text Generation with BERT," in *International Conference on Learning Representations*, 2020.

[9] M. K. Eddine, N. Tomeh, N. Habash, J. Le Roux, and M. Vazirgiannis, "AraBART: a Pretrained Arabic Sequence-to-Sequence Model for Abstractive Summarization," *arXiv preprint arXiv:2203.10945*, 2022.

[10] C.-Y. Lin, "ROUGE: A Package for Automatic Evaluation of Summaries," in *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, 2004, vol. 2, pp. 74–81 [Online]. Available: http://aclweb.org/anthology/W04-1013

[11] Y. Liu and M. Lapata, "Text Summarization with Pretrained Encoders," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019, pp. 3730–3740.

[12] M. Al-Maleh and S. Desouki, "Arabic Text Summarization Using Deep Learning Approach," *J Big Data*, vol. 7, no. 1, pp. 1–17, 2020.

[13] D. Suleiman and A. Awajan, "Multilayer Encoder and Single-Layer Decoder for Abstractive Arabic Text Summarization," *Knowl Based Syst*, vol. 237, p. 107791, 2022.

[14] K. Ganesan, C. Zhai, and J. Han, "Opinosis: A Graph-Based Approach to Abstractive Summarization of Highly Redundant Opinions," in *Proceedings of the 23rd International Conference on Computational Linguistics*, 2010, pp. 340–348.

[15] W. Etaiwi and A. Awajan, "SemG-TS: Abstractive Arabic Text Summarization Using Semantic Graph Embedding," *Mathematics*, vol. 10, no. 18, p. 3225, 2022.

[16] A. M. Al-Numai and A. M. Azmi, "The development of single-document abstractive text summarizer during the last decade," in *Trends and applications of text summarization techniques*, IGI Global, 2020, pp. 32–60.

[17] A. M. Al-Numai and A. M. Azmi, "Arabic text abstractive summarizer using ant colony system," Submitted for publication, 2023.

[18] A. M. Azmi and N. I. Altmami, "An Abstractive Arabic Text Summarizer with User Controlled Granularity," *Inf Process Manag*, vol. 54, no. 6, pp. 903–921, 2018.

[19] A. Azmi and S. Al-Thanyyan, "Ikhtasir—A User Selected Compression Ratio Arabic Text Summarization System," in *International Conference on Natural Language Processing and Knowledge Engineering*, 2009, pp. 1–7.

[20] A. M. Azmi and S. Al-Thanyyan, "A Text Summarizer for Arabic," *Comput Speech Lang*, vol. 26, no. 4, pp. 260–273, 2012.

[21] P. Qi, Y. Zhang, Y. Zhang, J. Bolton, and C. D. Manning, "Stanza: A Python Natural Language Processing Toolkit for Many Human Languages," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, 2020 [Online]. Available: https://nlp.stanford.edu/pubs/qi2020stanza.pdf

[22] H. Mubarak, "Build Fast and Accurate Lemmatization for Arabic," in *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, May 2018 [Online]. Available: https://aclanthology.org/L18-1181

[23] T. Zerrouki, "Tashaphyne, Arabic Light Stemmer." 2012 [Online]. Available: https://pypi.python.org/pypi/Tashaphyne/0.2

[24] K. Taghva, R. Elkhoury, and J. Coombs, "Arabic stemming without a root dictionary," in *International Conference on Information Technology: Coding and Computing (ITCC'05)-Volume II*, 2005, vol. 1, pp. 152–157.

[25] K. Abainia and H. Rebbani, "Comparing the effectiveness of the improved ARLSTem algorithm with existing Arabic light stemmers," in *2019 International Conference on Theoretical and Applicative Aspects of Computer Science (ICTAACS)*, 2019, vol. 1, pp. 1–8.

[26] K. Ganesan, "ROUGE 2.0: Updated and Improved Measures for Evaluation of Summarization Tasks." arXiv, 2015 [Online]. Available: https://arxiv.org/abs/1803.01937

[27] A. M. Azmi, M. N. Almutery, and H. A. Aboalsamh, "Real-Word Errors in Arabic Texts: A Better Algorithm for Detection and Correction," *IEEE/ACM Trans Audio Speech Lang Process*, vol. 27, no. 8, pp. 1308–1320, 2019.