EXTRACTIVE SUMMARIZATION FOR ARABIC DOCUMENTS USING SARABERT WITH A SEMANTIC SIAMESE SIMILARITY EVALUATION METRIC

Sami Shames El Deen , Mariette Awad
American University of Beirut

arXiv:2608.20964v1 [cs.CL] 21 Aug 2026

***Abstract*—In this research, we introduce SAraBERT, an enhanced version of AraBERT which proposes inter-sentence transformer layers for extractive summarization tasks. To ensure that the summaries generated by SAraBERT achieve a high coverage of the document's main ideas, we propose Semantic Siamese Similarity, a novel evaluation metric that measures the level of similarity between two text inputs. We validated using BLEU, ROUGE, and Semantic Siamese similarity on Sarabert and published related models. Simulation results showed the effectiveness of our proposed model and motivate follow on research.**

***Impact Statement*—This paper introduces SAraBERT, an improvement on Arabert, that incorporates inter-sentence transformer layers to perform extractive summarization on Arabic text. Furthermore, a novel called Semantic Siamese Similarity (SSS) is proposed to combine embedding similarity and exact match. Experiments using various evaluation metrics show that SAraBert outmatches its predecessors on Arabic extractive summarizing reaching new state of the art levels of performance.**

***Index Terms*—Arabic NLP , Text Summarization, Extractive Summarization, Transformers, Evaluation Metric, Sequence-to-Sequence Framework**

I. Introduction

As the amount of textual information available online grows rapidly, it becomes difficult for readers to read large amounts of text and find out which of these texts are useful. As a result, researchers in the field of automatic text summarization need tools that can make multiple-document reading more efficient. The task of text summarization is considered one of the most important and challenging NLP tasks. Summarization generates short text from long text, so the short text contains the most important information from the original text. The task is often divided into two paradigms known as extractive summarization and abstractive summarization. The first methodology determines the essential sections of the text using statistical tool. The summary is represented by truncating and connecting these sections. The second methodology emulates human activity in summarizing, which is based on presenting the text's core idea in a new linguistic style and using different terms. It incorporates more complicated procedures including paraphrase, generalization, and reordering [1].This work focuses on extractive summarization. While there is a wealth of research in the area of extractive summarization, most of this research is based on English texts, and there is a lack of research on summarizing Arabic texts.

Arabic natural language processing (NLP) [2] is considered more complex than English and other European languages. The main reason for this complexity is the highly derived and rich form of Arabic morphology. This creates a set of challenges, which include:

1) Morphological richness [3][4]: Arabic is heavily derived, inflected and has a significant impact on NLP tasks such as stemming and lemming.

2) Absence of diacritics: diacritics play an important role in determining the meaning of words and facilitating the task of tokenizing and parsing text.

3) No capitalization in Arabic language: without the usage of uppercase letters in Arabic, it will be difficult to identify proper nouns, titles, and abbreviations.

In this work, we introduce SAraBERT, a novel and enhanced version of AraBERT that adds inter-sentence transformer layers for extractive summarization tasks. To ensure that the summaries generated achieve a high coverage of the document's main ideas, we also propose a novel evaluation metric, Semantic Siamese Similarity that measures the level of similarity between two text inputs using contextualized embeddings in addition to exact match. We use an English benchmark dataset (CNN/Daily Mail) and translate it to Modern Standard Arabic language to train SAraBERT. Testing the performance of Sarabert in comparison with other published models was evaluated using BLEU, ROUGE, and Semantic Siamese similarity on Kalimat Dataset. Results have shown the effectiveness of our proposed model.

In summary, the main contributions of this work are as follows:

- A novel Arabic language model for extractive summarization that builds on AraBERT model.
- A novel hybrid similarity metric that measures similarities between 2 text based on the semantic and syntactic features.
- An arabic corpus translated from English that targets the extractive summarization and Question Answering tasks, available upon request.

The structure of the paper is such that, in section 2, we survey the related work and background information required for this work. Section 3 describes the model's architecture

and the proposed metric. Section 4 shows the experimental research and highlights the obtained results along with some insights. Finally Section 5 concludes the paper with follow on research directions.

II. Literature Review

To provide the necessary context to the proposed solution, it is worth investigating previous research.

A. Classical Approaches

Until 2017, statistical methods were predominantly used in text summarization which are based on the concept of relevance score and Bayesian classifiers [5]. Word Frequency approach is the most used methodology for sentence scoring where the sentence score is calculated by the sum of the frequencies while avoiding all stop words. The proposed solution in [6] generates news titles by combining Word-Frequency, sentence position and similarity measures methodologies. Term Frequency-Inverse Document Frequency (TF-IDF) is a numerical methodology that represents the importance of a word to document in a collection of documents (corps) [7]. TF-IDF is an improvement on the Word Frequency and shows how the weights are distributed on the words of a document. TF-IDF is used frequently in auto-summarization systems [8] [9] [10] [11]. An Arabic summarization system called AATSS [12] adopted an extractive text summarization approach that was mainly based on sentence weighting and scoring. However, it highly depends on the size of the document in terms of number of sentences, which leads to generation of improper grammar context, and it lacks automatic Arabic entity recognition system and Arabic pronoun resolution system. The classical statistical approaches are used for both single and multi-document summarization and can be used to enhance the selection of important sentences or the elimination of redundant sentences but fail to understand the text, since they only depend on statistical measures [13]. [14] proposed a solution for English language by tokenizing the text into clean sentences, then passing the sentences into a BERT model to generate embeddings. Then using K-means they generated clusters and selected summary sentences based on the closest sentence embedding to each cluster centroid.

B. Machine Learning Approach

Text summarization is considered as a binary classification problem, where a set of documents and their extractive summaries are used as a training set, and each sentence is classified as a summary sentence or non-summary based on statistical, semantic features or a combination of them [15] [16] [17] [18]. Machine learning methods can be suitable for document summarization as several methods have been introduced and proven to be effective in performing summarization tasks. We will discuss sequence to sequence models and encoder decoder architectures in what follows.

Sequence-to-Sequence Model

The neural abstractive summarization with sequence-to-sequence models emerged in [19] [20]. This approach has been applied to tasks such as headline generation [21] and article summarization [22]. Chopra et al. [23] show that attention approaches that are more specific to summarization can further improve the performance of models. Molham et al. [24] re-implemented the sequence-to-sequence framework on the Arabic language, which had not witnessed the employment of this model in the text summarization before. However, the authors state that the work still requires expanding the dataset to cover more articles, and to train new models that are beneficial with the Arabic language since it has a unique grammatical language written from right to left.

Encoder-Decoder Model

Google AI's pre-trained language model called Bidirectional Encoder Representations from Transformers (BERT), proved highly efficient at language understanding and achieved convincing results in most NLP tasks. The Arabic language model ARABERT based on BERT for Arabic language was evaluated on Named Entity Recognition, Sentiment Analysis and Question Answering [25]. Abdulla et al. [26] proposed an extractive Arabic text summarizer based on ARABERT to summarize the Arabic documents by evaluating and extracting the most important sentences at a document. This proposed approach generated a good summary by extracting the most important sentences from paragraphs. However, the model highly depends on sentence boundaries, its coverage accuracy decreases when the text is too long, and extracted sentences sometimes contain linguistic expressions that creates ambiguous summaries.

C. Other Approaches

Liu et al. of [27] managed to solve the problem of sentence boundary detection by adding an additional layer in the embedding that specifies the beginning and end of each consecutive sentence inserted into the encoder.

Go et al. [28] explored the effects of language variants, data sizes, and fine-tuning task types in Arabic pre-trained language models. The results suggested that the variant proximity of pre-training data to fine-tuning data is more important than the pre-training data size. Moreover, other design decisions can be explored that may contribute to the fine-tuning performance, including vocabulary size, tokenization techniques, and additional data mixtures.

Nasser et al. [29] presented an LSTM-based morphological disambiguation system for Arabic which has significantly outperformed the state-of-the-art systems. The paper suggested exploring additional deep learning architectures for morphological modeling and disambiguation, especially sequence-to-sequence models. It also suggested to further investigate the role of syntax features in morphological disambiguation, and explore additional techniques for more accurate tagging.

Reda et al. [30] developed a hybrid transformer based approach for Arabic text summarisation. First, they performed extractive summarization using a AraBert based sentence embedding approach to determine which sentences of an article are most suited to be part of the summary. Then, they performed abstractive summarization on the extracted summary by feeding it into the MT5 Arabic transformer. They

tested their model on the ESAC dataset and got a precision of 53%, a recall of 55%, and an f1 score of 49%. The main shortcoming of this work are as follows. First, pretrained models were used meaning that the models were not fine tuned specifically to perform summarization. Second, the ESAC dataset is limited to only 153 articles, and 700 summarizations. Using specialized models and a larger dataset may further improve the results.

III. Proposed Methodology

A. SAraBERT

Let d denote a document containing several sentences $[sent_1, sent_2, \ldots, sent_m]$, where $sent_i$ is the i^{th} sentence in the document. Extractive summarization can be defined as the task of assigning a label $y_i \in \{0, 1\}$ to each $sent_i$, indicating whether the sentence should be included in the summary or not. Thus, if $y_i = 1$ then the sentence is considered in the summary due to its importance in terms of document content. BERT as shown in Figure[1] is considered the model of choice because it showed better performance compared to other NLP statement embedding algorithms. To keep the original BERT pre-training objective, AraBERT uses Masked Language Modeling (MLM) to improve the pre-training process by letting the model predict the entire word rather than receiving clues from part of the word. In addition, it also uses Next Sentence Prediction (NSP) to help the model understand the relationships between sentences [25]. To achieve sentence ranking in the document based on content importance, we will apply modifications to the embedding layers as shown in Figure[2] to highlight sentence endpoints for the model to rank each sentence independently.

Encoding Multiple Sentences: We insert $[CLS]$ token before each sentence and a $[SEP]$ token after each sentence. The $[CLS]$ is used as a symbol to represent the start of a sequence and the $[SEP]$ indicated the end of that sequence. Inserting multiple $[CLS]$ tokens help in specifying the boundaries of each sentence.

Interval Segment Embedding: We add another layer of embedding to distinguish multiple sentences within a document. So, for $sent_i$ we assign a segment embedding E_A or E_B based on whether it's even or odd. For example, For $[sent_1, sent_2, sent_3, sent_4, sent_5]$ the interval segment layer assign $[E_A, E_B, E_A, E_B, E_A]$. The vector T_i that corresponds to the i^{th} $[CLS]$ token, will be used as the representation for $sent_i$.

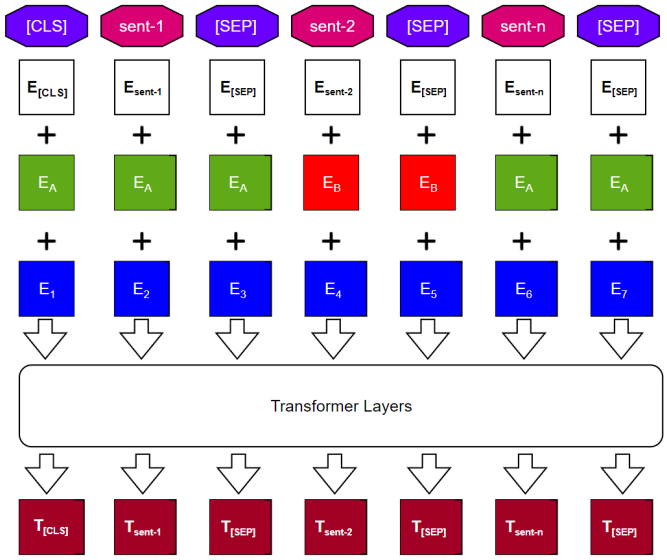

Fig. 1: Overview for the architecture of the original BERT model

The sequence on top is the input document, followed by the summation of three kinds of embeddings for each token. The summed vectors are used as input embeddings to the transformer layers, generating contextual vectors for each token T_i. The difference between SAraBERT and the model

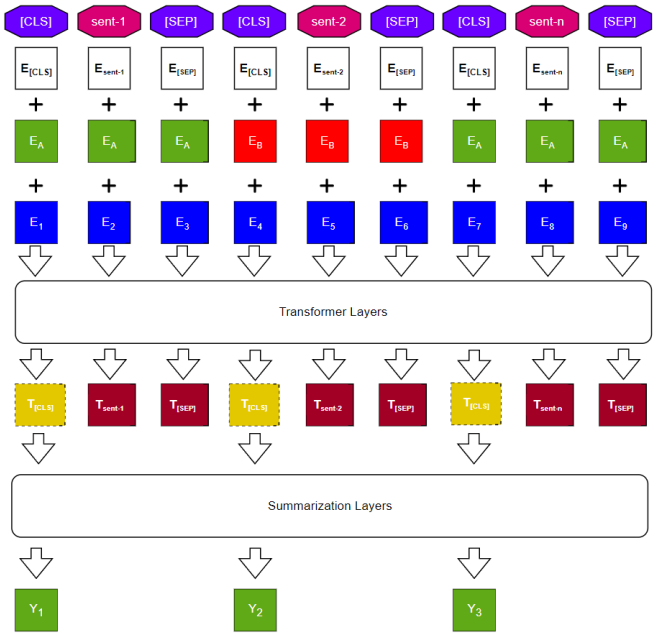

Fig. 2: Overview for the architecture of the SAraBERT model

in Figure[1] is the insertion of [CLS] tokens before each sentence, and alternating the sentence segmenting embedding values along with the addition of a summarization layer that derives a sentence score from the [CLS] token embeddings.

Algorithm 1 Summarization Model

Input: Text
Output : Sentence Scores

$tokens \leftarrow tokenize$(Text)
$sentences \leftarrow sentence\_segmentation(tokens)$
$tokenID \leftarrow array\,[|sentences|]\,[max_{sent \in sentences}(|sent|)]$
$segmentID \leftarrow array\,[|sentences|]\,[max_{sent \in sentences}(|sent|)]$
for $i = 1$ to $|sentences|$ **do**
  sentences[i] $\leftarrow$ {"[CLS]", sentences[i], "[SEP]"}
  for $j = 1$ to $|sentences[i]|$ **do**
    tokenID[i][j]$\leftarrow$ getTokenID(sentences[i][j])
    if i is Even **then**
      segmentID[i][j]$\leftarrow 0$
    else
      segmentID[i][j]$\leftarrow 1$
    end if
  end for
end for
bertEmbedding $\leftarrow$ AraBERT(tokenID,segmentID)
sentence_embeddings $\leftarrow$ get_CLS_embeddings(bertEmbedding)
sentence_scores$\leftarrow$ encode(sentence_embeddings)

After obtaining the embeddings of CLS tokens of each sentence, the tokens will get fed into an encoder $\hat{Y} = H(T)$ where H is the encoding model, T is a vector of CLS embeddings, and $\hat{Y}$is a scoring vector with $\hat{y}_i$ representing the score of the i^{th} sentence. The model H has been interchanged between Multi-Layer Perceptrons (MLP), Recurent Neural Network (RNN), and Transformers to compare the different scoring criteria each encoder have. Results are shown in section V

B. Siamese Semantic Similarity (SSS)

ROUGE [31] has been used as a metric for determining the quality of a summary by comparing a candidate summary (generated by machine) to a reference summary (generated by humans). The way it does this projects only the level of similarity on the syntactic level without the coverage of the context. Summaries may cover a large portion of the original context but with fewer words (and more verbose) which Rouge can't keep track of. Language models have the ability to map the context of a passage into a fixed size vector. We can compute the cosine similarity of the 2 embeddings to obtain a similarity measure at the semantic level.

TABLE I: An example comparing the ROUGE score and Cosine Similarity

Reference	علي يأكل الطعام في الليل	ROUGE	Cosine Similarity
Candidate	هو يتناول الطعام مساء	0.0	0.9074

Since cosine similarity treats all dimensions equally in the semantic space, and 2 vector points might overlap yet stay very distant from each other, the distance should be added into the evaluation, since the smaller the value is the higher the score should be, we insert the distance difference in the denominator. Moreover, we will wrap it with a square root to slow the growth speed of the value as the difference increase (this will later become helpful in interpreting the computed values).

$$\frac{cos\_sim(cand_{embedding}, ref_{embedding})}{\sqrt{||cand_{embedding} - ref_{embedding}||_2 + 1}} \tag{1}$$

A value of 1 was added because the distance difference can be 0, thus to eliminate the possibility of a division by zero error.

TABLE II: An example showing the cosine similarity and L2 norm when comparing a reference to predictions

Reference	تناول الولد تفاحة	Cos_Sim	L2 norm
Candidate$_1$	أكل الطفل قطة	0.873	5.989
Candidate$_2$	أكل الطفل فاكهة	0.867	5.686

Moreover, ROUGE is still an important component of the formula to keep an eye on the grammar. An average of ROUGE 1, ROUGE 2, and ROUGE L is used.

$$\frac{cos\_sim(cand_{embedding}, ref_{embedding}) \cdot rouge(cand, ref)}{\sqrt{||cand_{embedding} - ref_{embedding}||_2 + 1}} \tag{2}$$

Finally, frequency should be considered. Two sentences can be similar in context yet one being redundant more than the other.

TABLE III: An example showing the cosine similarity, ROUGE score, L2 norm, and frequency difference when comparing a reference to predictions

تناول الولد تفاحة	Cos_Sim	Rouge	L2 norm	FreqDiff
أكل الطفل تفاحة	0.957	0.222	3.637	1.000
الولد تفاحة تفاحةتناول تناول	0.930	0.889	5.005	1.667

Thus a penalty should be added on the difference in frequency distribution among words where we divide the the number of unique words over the number of total words. This step was added because an embedding does not take redundancy and coverage into consideration which can affect level of similarity for tasks such as summarization. The final Formula can be seen in Algorithm[2]

TABLE IV: interpretation of SSS scores

SSS score	Interpretation
0-0.1	no clear resemblance or no similarity at all
0.1-0.2	the gist is clear, but not all context is covered
0.2-0.5	context is significantly covered using different words
$\geq$0.5	exact replica with slight modifications

Algorithm 2 Semantic Textual Similarity

Input: Candidate, Reference
Output : SimilarityScore

$rouge_1 \leftarrow rouge_1(\text{Candidate}, \text{Reference})$
$rouge_2 \leftarrow rouge_2(\text{Candidate}, \text{Reference})$
$rouge_L \leftarrow rouge_L(\text{Candidate}, \text{Reference})$
$\text{rouge} \leftarrow \frac{rouge_1 + rouge_2 + rouge_L}{3}$
$embedding_1 \leftarrow AraBERT.encode(\text{Candidate})$
$embedding_2 \leftarrow AraBERT.encode(\text{Reference})$
$cosine\_similarity \leftarrow \frac{embedding_1 \cdot embedding_1}{\|embedding_1\|_2 \times \|embedding_2\|_2}$
$DistDiff \leftarrow \|embedding_1 - embedding_2\|_2$
$FreqDiff_c \leftarrow \frac{unique\_words(Candidate)}{total\_words(Candidate)}$
$FreqDiff_r \leftarrow \frac{unique\_words(Reference)}{total\_words(Reference)}$
$FreqDiff \leftarrow \frac{min(FreqDiff_r, FreqDiff_c)}{max(FreqDiff_r, FreqDiff_c)}$
$\text{SimilarityScore} \leftarrow \frac{\text{cosine\_similarity} \times \text{rouge} \times FreqDiff}{\sqrt{DistDiff+1}}$

IV. Experiment

A. Dataset

1) CNN/ Daily Mail: We used the CNN / DailyMail dataset to train SAraBERT, it is an English dataset containing over 300,000 unique news articles written by CNN and DailyMail journalists. The dataset can be used to train a model for the task of extractive and abstractive summarization. Each document instance from the dataset consists of 3 components:

1) **id**: A string containing the hexadecimal SHA1 hash of the URL from which the story was obtained.
2) **Article**: A string containing the body of a news article.
3) **Highlights**: A string containing the highlights of the article written by the author of the article.

Since our model should be trained on an Arabic dataset, we examined state-of-the-art machine translation model mbart [32] along with Google-Trans[1] service. We evaluated each of the models on the Opus dataset [32]. Results are shown in Table[V]. We can see competitive results from both translators, so the tie breaker was the duration required to translate a document.

TABLE V: Translation Experiment Results

	BLEU	METEOR	Duration (Docs/minute)
GoogleTrans	0.413 ± 0.386	0.24	30
mBart	0.413 ± 0.382	0.25	20

We also generated diacriticisation for the Arabic documents [33] to see whether the diacritics have an effect on the readability level or not. Table[VI] and Figure[4] shows that the documents containing diacritics have closer readability estimations to English than documents without diacritics but it will not be a problem since [34] explained that this behavior is normal and what is important is that the correlation between Flesch and OSMAN (without diacritics) is high.
We investigated if translation will affect the readability of a document and whether large difference between the English

[1]pypi.org/googletrans

Fig. 3: Sample English passage translated to Arabic and Diactritized

and Arabic readabilities of the original and translated documents will influence our model. After experimenting on 1000 translated samples, and computing the ROUGE score of the resulting summaries along with the readability of the original and translated samples, results have shown that the level of readability does not affect the model's output and there is no correlation between readability difference and ROUGE. More details are provided in Appendix[VII-C].

TABLE VI: Average Readability Measures for Arabic

Metric	Value (Mean ± std)
OSMAN - Diacritics	78.45±5.18
OSMAN + Diacritics	70.28±8.95

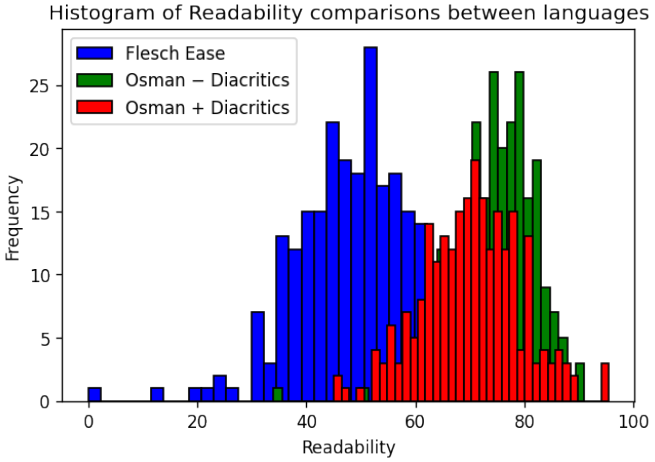

Fig. 4: Histogram of readability comparisons between english and arabic with and without diacritics

V. Results

Results are reported in Table[IX]. The 3 metrics used for evaluation are BLEU (Average of the 4 BLEU evaluations over uni-,bi-,tri-, and quad-grams), ROUGE (Average between F1-Scores of ROUGE-1,ROUGE-2, and ROUGE-L), and the Siamese Similarity metric. Oracle Score represents the evaluation of human summaries that will be used for comparison with machine summaries. "SAraBert+RNN" was the best performing model based on the 3 evaluation metrics. Sample summaries given by the different models can be found in the Appendix[VII-B].

TABLE VII: BLEU scores over Kalimat Dataset

Model	bleu_1	bleu_2	bleu_3	bleu_4	avg	min	max
SAraBert + Classifier	0.408	0.385	0.38	0.376	0.387	0.0	0.99
SAraBert + RNN	**0.474**	**0.45**	**0.445**	**0.441**	**0.452**	0.0	0.99
SAraBert + Transformer	0.391	0.368	0.364	0.36	0.371	0.0	0.99
AraBert Embeddings + K-Means	0.284	0.254	0.249	0.246	0.258	0.0	0.991
AraVec + K-Means	0.345	0.334	0.331	0.328	0.335	0.0	0.973
Bag of Words	0.361	0.351	0.348	0.345	0.352	0.0	0.98

TABLE VIII: ROUGE scores over Kalimat Dataset

Model	rouge_1	rouge_2	rouge_L	avg	min	max
SAraBert + Classifier	0.529	0.466	0.51	0.502	0.008	0.99
SAraBert + RNN	**0.588**	**0.524**	**0.568**	**0.56**	0.008	0.99
SAraBert + Transformer	0.512	0.444	0.491	0.482	0.017	0.99
AraBert Embeddings + K-Means	0.445	0.366	0.427	0.414	0.008	1.0
AraVec + K-Means	0.545	0.49	0.535	0.523	0.0	1.1
Bag of Words	0.47	0.415	0.46	0.448	0.0	1.1

TABLE IX: Results of different models under several evaluation metrics

Model	BLEU	ROUGE	Siamese Similarity	BS	TS-SS
Oracle Score	0.532 ± 0.27	0.699 ± 0.23	0.326 ± 0.3	0.83	0.003
SAraBert + Classifier	0.387 ± 0.3	0.502 ± 0.31	0.211 ± 0.158	0.76	0.018
SAraBert + RNN	**0.452** ± 0.2	**0.560** ± 0.26	**0.282** ± 0.18	**0.79**	0.017
SAraBert + Transformer	0.371 ± 0.3	0.482 ± 0.31	0.240 ± 0.17	0.77	0.019
AraBert Embeddings + K-Means	0.258 ± 0.21	0.414 ± 0.25	0.216 ± 0.1	0.76	0.032
AraVec + K-Means	0.335 ± 0.28	0.523 ± 0.28	0.191 ± 0.18	0.74	0.039
Bag of Words	0.352 ± 0.35	0.448 ± 0.36	0.217 ± 0.28	0.76	0.111

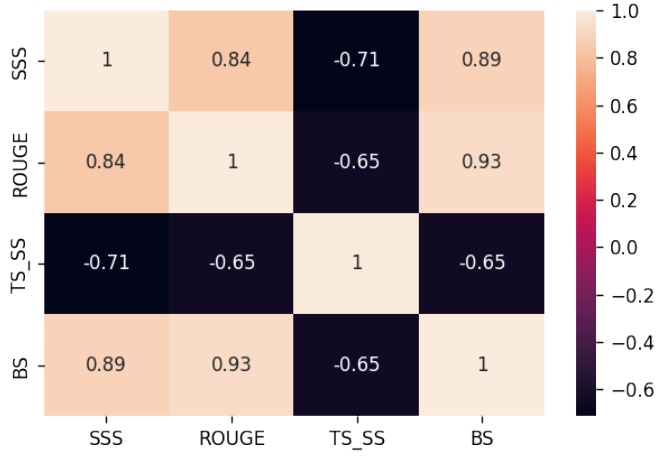

Fig. 5: Pearson Correlation between the similarity metrics

VI. Discussion

It appears that the usage of BiLSTM encoder did better than the transformer encoder, this could be due to the need of transformers for huge amount of data to train all its attention heads and layers. Another interesting thing to notice is that Bag of Words method got a high score on the Siamese Similarity distinguishably higher than the other classical approaches. One problem remains is the incapability of feeding very large documents into the models to obtain a single global lookup on the document instead of segmenting the document and loosing linked contexts between the trimmed passages. Future work will focus on the SSS metric, where the cosine similarity should give more importance/weight/attention to specific context features.

VII. Conclusion

Text summarization is one of the important areas of research in NLP, as textual data keep increasing each day. The proposed model "*SAraBert*" works on the summarization task for MSA NLP. We applied experiments on a large-scale translated dataset and found that SAraBert with RNN encoding layers can achieve the best performance. We also created a similarity metric that evaluated the similarity of 2 documents on the semantic level and not just the syntactic level.

Appendix

A. Siamese Semantic Similarity Experiments

The metric that was discussed is Section[III-B] had to be tested for different cases to visualise and evaluate the overall performance of this metric.

We applied SSS on a set of candidates shown in Table[X] to the reference found at the first row of that table. The results of SSS and BertScore can be found in Table[XI] along with the values required to compute SSS. The correlation between BertScore and SSS is 0.97 showing that both metrics behave very similar but SSS tends to be more strict with its evaluations.

TABLE XI: Results of computing SSS and BertScore (BS) along with other required computations for the candidates in Table[X] linked by ID

ID	Cosine	ROUGE	norm_2	FD	SSS	BS
1	0.598	0.0	9.406	0.923	0.0	0.631
2	0.638	0.111	9.1	0.923	0.065	0.682
3	1.0	1.0	1.0	1.0	1.0	1.0
4	0.995	0.909	2.012	1.0	0.901	0.971
5	0.980	0.857	2.929	0.993	0.829	0.935
6	0.877	0.41	5.719	0.923	0.329	0.772
7	0.734	0.205	8.86	0.923	0.137	0.697
8	0.816	0.286	6.894	0.923	0.213	0.728
9	0.7630	0.19	7.789	0.923	0.133	0.693
10	0.8000	0.19	7.526	0.923	0.139	0.718
11	0.763	0.19	7.926	0.923	0.133	0.702
12	0.824	0.0	6.756	0.923	0.0	0.717
13	0.891	0.238	5.396	0.923	0.194	0.74
14	0.812	0.314	7.452	0.923	0.233	0.722
15	0.904	0.314	5.205	0.923	0.26	0.741
16	0.701	0.0	8.136	0.923	0.0	0.662
17	0.811	0.058	6.777	0.923	0.043	0.669
18	0.722	0.078	8.0	0.923	0.052	0.623
19	0.845	0.257	6.247	0.923	0.199	0.743
20	0.677	0.286	9.243	0.217	0.041	0.616
21	0.869	0.874	6.778	0.433	0.326	0.778

B. Sample Summaries form SaraBert

This section will present sample summaries extracted via SaraBert, highlights are used to visualize the selected top 3 sentences selected to be considered as the most informative sentences.

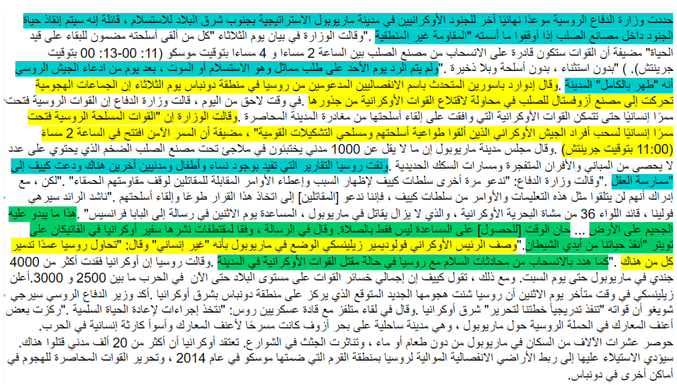

Fig. 6: Sample SaraBert summarization. Yellow highlighting represents summary extraction using MLP as encoder, Green represents RNN encoder, and Blue represents Transformer

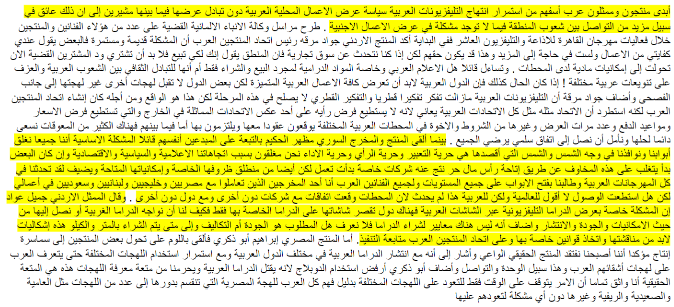

Fig. 7: Sample SaraBert+BiLSTM of a good summarization

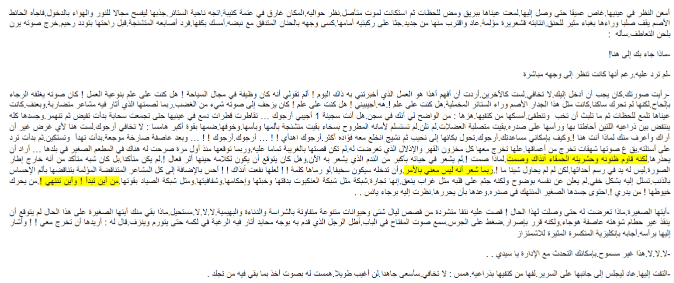

Fig. 8: Sample SaraBert+BiLSTM of a bad summarization

C. Translation Quality

In this section we will present sample translations and discuss the quality and performance effect on the results.

A Georgia man is heading to prison for nearly 10 years for his role in a credit card fraud gang associated with a website called Carder.su that is linked to $50 million in worldwide losses. Cameron Harrison, aka "Kilobit," 28, of Augusta, Georgia, pleaded guilty to possessing more than 260 compromised credit and debit card numbers, which were recovered from his computer and email accounts following his arrest. He also admitted, according to the Department of Justice, that the Carder.su organization committed money laundering, narcotics trafficking and computer crimes. He said members tried to avoid detection by communicating through various encrypted forums, such as chat rooms, private messaging systems and virtual private networks. Harrison was identified when he purchased a counterfeit Georgia driver's license from an undercover special agent through the Carder.su network. During interactions with the undercover special agent, Harrison admitted he had been a vendor of counterfeit identifications in the defunct cyberfraud organization "ShadowCrew." Fifty-five individuals were charged in four separate indictments in Operation Open Market, run by Homeland Security Investigations and the U.S. Secret Service. To date, 26 individuals have been convicted and the rest are either fugitives or are awaiting trial. Harrison pleaded guilty to participating in a racketeer influenced corrupt organization, conspiracy to engage in a racketeer influenced and corrupt organization, and trafficking in and production of false identification documents. "This significant sentence is entirely fitting given that this defendant's actions and those of the larger criminal organization harmed countless innocent Americans and seriously compromised our financial system," said Homeland Security Investigations Executive Associate Director Peter T. Edge. "Criminals like this defendant who believe they can elude detection by hiding behind their computer screens here and overseas are discovering that cyberspace affords no refuge from American justice."

Fig. 9: Sample English passage with Fleach score of 11.38 (Hard to read)

يتوجه رجل جورجيا إلى السجن لمدة 10 سنوات تقريبا لدوره في عصابة احتيال ببطاقة الائتمان المرتبطة بموقع إلكتروني يسمى Carder.SUالمرتبط بمبلغ 50 مليون دولار في الخسائر في جميع أنحاء العالم .أقر كاميرون هاريسون، المعروف أيضا باسم "كيلوبيت،" 28، من أوغستا، جورجيا، بالذنب على امتلاك أكثر من 260 أرقام بطاقات ائتمان وخصم الخصم، والتي تم استردادها من أجهزة الكمبيوتر الخاصة به وحسابات البريد الإلكتروني بعد اعتقاله .كما اعترف، وفقا لوزارة العدل، أن منظمة الكاردر. ارتكبت غسل الأموال واتجار المخدرات وجرائم الكمبيوتر. وقال إن الأعضاء حاولوا تجنب الكشف عن طريق الاتصال من خلال مختلف المنتديات المشفرة، مثل غرف الدردشة وأنظمة المراسلة الخاصة والشبكات الخاصة الافتراضية .تم تحديد هاريسون عندما اشترى رخصة قيادة جورجيا المزيفة من وكيل خاص سري من خلال شبكة .Carder.Su أثناء التفاعلات مع الوكيل الخاص السريري، اعترف هاريسون بأنه بائع للترديدات المزيفة في منظمة سيتيرفراود بتفين ."Shadowcrecrew" تم اتهام خمسة وخمسين فردا في أربع لوائح اتهام منفصلة في عملية السوق المفتوحة، وتشغيلها من خلال تحقيقات الأمن الداخلي والخدمات السرية الأمريكية. حتى الآن، تم إدانة 26 شخصا والباقي إما الهاربين أو ينتظرون المحاكمة .أقر هاريسون بأنه مذنب للمشاركة في مضرب يؤثر على المنظمة الفاسدة، والتآمر للانخراط في مضرب يتأثر ونظام فاسد، والاتجار بمستندات الهوية الخاطئة وإنتاجها .وقال مدير الأمن الداخلي للسيارات التنفيذية بيتر تي إيدج "هذه الجملة الهامة تترافق تماما بالنظر إلى أن أفعال المدعى عليه هذه وأكبر من المنظمة الإجرامية الأكبر التي أضرت عدد لا يحصى من الأميركيين الأبرياء الذين لا حصر لهم الذين يعانون من نظامنا المالي." "مجرمو مثل هذا المدعى عليه الذين يعتقدون أنهم يستطيعون الاكتشاف عن طريق الاختباء وراء شاشات الكمبيوتر هنا والخارج لا يكتشفون أن الفضاء الإلكتروني لا يملأ من العدالة الأمريكية."

Fig. 10: Sample Arabic passage (translation of Figure[9]) with Osman score of 66.73 (Slight hard to read)

TABLE X: List of a reference sentence (1) and set of candidate sentences(2 to 19)

ID	Text
0	أكل التلميذ سامي طعامه في المدرسة صباحا يوم الثلاثاء. طعامه كان تفاحة
1	هرب
2	أكل
3	أكل التلميذ سامي طعامه في المدرسة صباحا يوم الثلاثاء. طعامه كان تفاحة
4	تناول التلميذ سامي طعامه في المدرسة صباحا يوم الثلاثاء. طعامه كان تفاحة
5	تناول سامي طعامه في المدرسة صباحا يوم الثلاثاء. طعامه كان تفاحة
6	أكل التلميذ سامي طعامه في المدرسة صباحا يوم الثلاثاء. طعامه كان تفاحة
7	أكل سامي
8	أكل سامي تفاحة
9	أكل سامي دجاجة
10	أكل سامي طعاما
11	أكل التفاح سامي
12	أكلت سامية طعامها
13	في المدرسة تناول سامي وجبة تفاح
14	في المدرسة أكل سامي
15	نهار الثلاثاء صباحا أكل سامي تفاحة
16	فازت الممثلة جيسيكا بجائزة اوسكار
17	كان يصوم معها حتى فهم أن ذلك يعني أنه لا يستطيع الأكل
18	اسمح لي أن أساعدك في أمتعتك
19	استيقظ سامي صباح يوم الثلاثاء متحمسا للذهاب الى المدرسة لانه يريد ان يتناول تفاحة هناك
20	أكل سامي تفاحة أكل سامي تفاحة أكل سامي تفاحة أكل سامي تفاحة
21	أكل أكل أكل أكل أكل التلميذ سامي سامي طعامه المدرسة المدرسة في المدرسة المدرسة صباحا يوم الثلاثاء يوم يوم يوم. طعامه كان تفاحة تفاحة تفاحة تفاحة تفاحة

When Grandma Nelly lays down a challenge, you'd better accept it. And she wasn't shooting low. Nelly was gunning for NBA star Dwyane Wade. "Dwyane Wade," she says, pointing her finger at the camera in a YouTube video. "On my 90th birthday, I want to play one-(on)-one with you." On Tuesday, she got her wish. Grandma Nelly -- AKA Illuminada Magtoto -- met Wade at the Heat practice facility for a little shoot-around. "I want to play with you," said the grinning grandma who barely came up chest-high to the 6-foot, 4-inch guard. "Now I'm 90." Wade appeared to be just as touched by the encounter. "This gives me life. This gives me a purpose," he said after it was over. Wade sealed the date with a kiss on the hand. "Oh my God," she squealed. "I'm very, very happy. I'm very grateful," Nelly said. "This is my dream come true."

Fig. 11: Sample English passage with Fleach score of 83.69 (Easy to read)

عندما تضع الجدة نيللي تحديا، من الأفضل قبولها. وكانت لا تطلق النار منخفضة. نيللي كان يطيح ل NBA Star Dwyane Wade. "Dwyane Wade"، تقول، تشير إلى إصبعها على الكاميرا في فيديو يوتيوب. "في عيد ميلادي التسعين، أريد أن ألعب (on) - One- معك." يوم الثلاثاء، حصلت على رغبتها.Grandma Nelly - AKA Illumaraada Magtoto - التقى واد في منشأة الممارسة الحرارية لقليل تبادل لاطلاق النار حولها. "أريد أن ألعب معك". "الآن أنا 90." بدا أن واد أن تكون مؤكدا تماما. وقال بعد انتهاء الأمر "هذا يعطيني الحياة. هذا يعطيني غرضا". أغلق واد التاريخ مع قبلة في اليد. "يا إلهي". وقال نيللي "أنا سعيد جدا جدا. أنا ممتن للغاية". "هذا حلمي حقيقة".

Fig. 12: Sample Arabic passage (translation of Figure[11]) with Osman score of 88.01 (Easy to read)

The translations have shown no effect on the trained model, where the readability variation had no link with ROUGE results. We sampled 1000 documents and computed their respective readability metrics in Arabic and English (Osman and Flesch) along with the ROUGE value of the generated summary. By looking at table[XIII], it can be observed that no readability metric highly affects the ROUGE evaluation.

TABLE XIII: Correlation between the different features

Index	Flesch	Osman	Rouge-AR	Readability-Diff
Flesch	1.0	0.76	-0.12	-0.64
Osman	0.76	1.0	-0.04	0.01
rouge-AR	-0.12	-0.04	1.0	0.14
Readability-Diff	-0.64	0.01	0.14	1.0

TABLE XII: Average value for documents computed over 1000 sample documents

Flesch	59.93 ± 10.13
Osman	74.88 ± 7.79
Rouge-AR	0.41 ± 0.14
Readability-Diff	14.95 ± 6.55

REFERENCES

[1] H. Jing, "Using hidden markov modeling to decompose human-written summaries," *Computational linguistics*, vol. 28, no. 4, pp. 527–543, 2002.

[2] N. Hajj and M. Awad, "Weighted entropy cortical algorithms for isolated arabic speech recognition," IEEE, 2013.

[3] A. B. Al-Saleh and M. E. B. Menai, "Automatic arabic text summarization: a survey," *Artificial Intelligence Review*, vol. 45, no. 2, pp. 203–234, 2016.

[4] K. S. Al Harazin, "Multi-document arabic text summarization," 2015.

[5] V. Patil, M. Krishnamoorthy, P. Oke, and M. Kiruthika, "A statistical approach for document summarization," *Department of Computer Engineering Fr. C. Rodrigues Institute of Technology, Vashi, Navi Mumbai, Maharashtra, India*, 2004.

[6] F. Alotaiby, S. Foda, and I. Alkharashi, "New approaches to automatic headline generation for arabic documents," *Journal of Engineering and Computer Innovations*, vol. 3, no. 1, pp. 11–25, 2012.

[7] R. Ferreira, L. de Souza Cabral, R. D. Lins, G. P. e Silva, F. Freitas, G. D. Cavalcanti, R. Lima, S. J. Simske, and L. Favaro, "Assessing sentence scoring techniques for extractive text summarization," *Expert systems with applications*, vol. 40, no. 14, pp. 5755–5764, 2013.

[8] Q. Al-Radaideh and M. Afif, "Arabic text summarization using aggregate similarity," in *International Arab conference on information technology (ACIT2009), Yemen*, 2009.

[9] A. Haboush, M. Al-Zoubi, A. Momani, and M. Tarazi, "Arabic text summarization model using clustering techniques," *World of Computer Science and Information Technology Journal (WCSIT) ISSN*, pp. 2221–0741, 2012.

[10] F. El-Ghannam and T. El-Shishtawy, "Multi-topic multi-document summarizer," *arXiv preprint arXiv:1401.0640*, 2014.

[11] H. N. Fejer and N. Omar, "Automatic arabic text summarization using clustering and keyphrase extraction," in *Proceedings of the 6th International Conference on Information Technology and Multimedia*, pp. 293–298, IEEE, 2014.

[12] N. M. Hewahi and K. A. Kwaik, "Automatic arabic text summarization system (aatss) based on semantic features extraction," *International Journal of Technology Diffusion (IJTD)*, vol. 3, no. 2, pp. 12–27, 2012.

[13] M. El-Haj, U. Kruschwitz, and C. Fox, "Multi-document arabic text summarisation," in *2011 3rd Computer Science and Electronic Engineering Conference (CEEC)*, pp. 40–44, IEEE, 2011.

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

[15] M. A. Fattah and F. Ren, "Ga, mr, ffnn, pnn and gmm based models for automatic text summarization," *Computer Speech & Language*, vol. 23, no. 1, pp. 126–144, 2009.

[16] R. Belkebir and A. Guessoum, "A supervised approach to arabic text summarization using adaboost," in *New contributions in information systems and technologies*, pp. 227–236, Springer, 2015.

[17] Q. A. Al-Radaideh and D. Q. Bataineh, "A hybrid approach for arabic text summarization using domain knowledge and genetic algorithms," *Cognitive Computation*, vol. 10, no. 4, pp. 651–669, 2018.

[18] A. Nenkova and K. McKeown, "A survey of text summarization techniques," in *Mining text data*, pp. 43–76, Springer, 2012.

[19] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[20] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in neural information processing systems*, pp. 3104–3112, 2014.

[21] A. M. Rush, S. Chopra, and J. Weston, "A neural attention model for abstractive sentence summarization," *arXiv preprint arXiv:1509.00685*, 2015.

[22] R. Nallapati, B. Zhou, and M. Ma, "Classify or select: Neural architectures for extractive document summarization," *arXiv preprint arXiv:1611.04244*, 2016.

[23] S. Chopra, M. Auli, and A. M. Rush, "Abstractive sentence summarization with attentive recurrent neural networks," in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 93–98, 2016.

[24] M. Al-Maleh and S. Desouki, "Arabic text summarization using deep learning approach," *Journal of Big Data*, vol. 7, no. 1, pp. 1–17, 2020.

[25] W. Antoun, F. Baly, and H. Hajj, "Arabert: Transformer-based model for arabic language understanding," *arXiv preprint arXiv:2003.00104*, 2020.

[26] A. M. Abu Nada, E. Alajrami, A. A. Al-Saqqa, and S. S. Abu-Naser, "Arabic text summarization using arabert model using extractive text summarization approach," 2020.

[27] Y. Liu, "Fine-tune bert for extractive summarization," *arXiv preprint arXiv:1903.10318*, 2019.

[28] G. Inoue, B. Alhafni, N. Baimukan, H. Bouamor, and N. Habash, "The interplay of variant, size, and task type in arabic pre-trained language models," 2021.

[29] N. Zalmout and N. Habash, "Don't throw those morphological analyzers away just yet: Neural morphological disambiguation for arabic," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 704–713, 2017.

[30] A. Reda, N. Salah, J. Adel, M. Ehab, I. Ahmed, M. Magdy, G. Khoriba, and E. H. Mohamed, "A hybrid arabic text summarization approach based on transformers," in *2022 2nd International Mobile, Intelligent, and Ubiquitous Computing Conference (MIUCC)*, pp. 56–62, 2022.

[31] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," in *Text summarization branches out*, pp. 74–81, 2004.

[32] Y. Tang, C. Tran, X. Li, P.-J. Chen, N. Goyal, V. Chaudhary, J. Gu, and A. Fan, "Multilingual translation with extensible multilingual pretraining and finetuning," *arXiv preprint arXiv:2008.00401*, 2020.

[33] Z. Barqawi, "Shakkala, arabic text vocalization," 2017.

[34] M. El-Haj and P. E. Rayson, "Osman: A novel arabic readability metric," 2016.