Cross-lingual Fine-tuning for Abstractive Arabic Text Summarization

Mram Kahla and **Zijian Győző Yang** and **Attila Novák**
Pázmány Péter Catholic University
Faculty of Information Technology and Bionics
Práter u. 50/a, 1083 Budapest, Hungary
{lastname.firstname(.midname)}@itk.ppke.hu

Abstract

While abstractive summarization in certain languages, like English, has already reached fairly good results due to the availability of trend-setting resources, like the CNN/Daily Mail dataset, and considerable progress in generative neural models, progress in abstractive summarization for Arabic, the fifth most-spoken language globally, is still in baby shoes. While some resources for extractive summarization have been available for some time, in this paper, we present the first corpus of human-written abstractive news summaries in Arabic, hoping to lay the foundation of this line of research for this important language. The dataset consists of more than 21 thousand items. We used this dataset to train a set of neural abstractive summarization systems for Arabic by fine-tuning pre-trained language models such as multilingual BERT, AraBERT, and multilingual BART-50. As the Arabic dataset is much smaller than e.g. the CNN/Daily Mail dataset, we also applied cross-lingual knowledge transfer to significantly improve the performance of our baseline systems. The setups included two M-BERT-based summarization models originally trained for Hungarian/English and a similar system based on M-BART-50 originally trained for Russian that were further fine-tuned for Arabic. Evaluation of the models was performed in terms of ROUGE, and a manual evaluation of fluency and adequacy of the models was also performed.

1 Introduction

When we talk about text summarization that practically means that using certain algorithms, we teach a machine to subtract information from an extensive text and provide a significantly shorter overview of it. And just like in the case of human beings, like a number of cases in our own school experiences, there are two ways of doing that.

The first way is called extractive summarization (Nallapati et al., 2016; Zhang et al., 2018; Narayan et al., 2018). In this method the idea is to practically highlight, take out certain keywords, phrases, or sentences from the text and put them together. Therefore, the result, the summary will use the exact same words, terms, and sentences as the original, and almost certainly even in the same order. It is practically the same method, when we humans glance through a massive text, like a thick script, or a book in just a few minutes. The machine would, just like our mind, focus on the first few words, paragraph, or page, then pick up the most commonly occurring words, and extract complete sentences with them, without changing anything in the sentence.

The second way is abstractive summarization (See et al., 2017; Paulus et al., 2017; Rush et al., 2015). Once again, this is not something new for our brain. Like in so many of our school studies, when a certain assignment was given to us and then later asked about in school, we probably did not use the same words and phrases of the original reading material, but we had a general idea about it, which we explained in our own plain words. And since our limited capability to remember and the inherent tendency for laziness, our summary was usually short, only an essence reflecting the original passage. Quintessentially that is abstractive summarization. Creative reconstruction of a textual message built on its comprehension.

We chose to focus on the Arabic language because it presents us challenges, which once surpassed can open up new fields of research.

Arabic has many features. One of the advantages of the Arabic language is that besides its huge variety of dialects and spoken versions - which is a naturally occurring phenomenon - its formal written version (Fusha), a practically dead language, is highly standardized and lacks major regional vari-

Proceedings of Recent Advances in Natural Language Processing, pages 655–663
Sep 1–3, 2021.
https://doi.org/10.26615/978-954-452-072-4_074

ety. But it still has a massive amount of primary users - native Arabic speakers - and text providers, also a big amount of secondary users, meaning people who are not native speakers themselves but in their engagements with Arab people, or organizations create new texts. This sort of lingual stability is rare among ”big languages”. So that would be the good side that we have a fairly standard massive corpus with minimal presence of dialectical variables complicating the learning process. On the other hand, Arabic is a language in which short vowels are not written. Though possible, usually not even marked in texts, therefore it is required for the reader to have extensive knowledge of the language, otherwise, the reader would not comprehend the message of the letters, even if he/she knows them.

For better comprehension let us see an example. In English, a given word, like ”wish” is always written and read the same way. It might be understood as a verb, or an identically written noun, so there is a number of variables we can attach to the word but given the surrounding text that is easy to process. With some words, like ”read”, which can be either past or present, and the application of idioms, this number of variables grows, but not significantly. And the number of words presenting such a feature is also fairly limited, as these are rather highlightable exceptions than rules. In Arabic, however, a three-letter word k-t-b, can automatically present 3 distinct forms, namely ”he wrote”, ”it was written” and ”books”. With minimal alteration, the number of possible solutions goes up to 20, or so, and that is a base rule with very few exceptions. And here we are still not talking about an agglutinative language, like Hungarian and Turkish, where the suffixes with a big number of variables, but cutting them the core stays fairly the same. With Arabic, we face a massive amount of inherent variables, all to be taken into consideration upon processing. The problem, however, presents an opportunity, as we can exploit this phenomenon to achieve bigger coherence in the text digestion.

Arabic also has another advantage, both scientifically and in the sense of application. It is one of the only 6 U.N. official languages, along with English, French, Spanish, Russian, and Chinese. That means that we have a massive resource of scientific and checked texts, beyond the usual quantity, upon which can be built, and which can be a potential test ground for further development. Meaning that we not only have a large quantity of informal, or semi-formal text between native speakers, but we also have a huge reviewed linguistically double-checked text. Given it is a U.N. language, massive amount of texts, which otherwise would have necessarily not concerned the Arab world, are translated into it and linguistically checked by professionals. With the inclusion of the political and economic value of the region, and the amount of politically sensitive and important material to be assessed, the value of a text summarization tool for Arabic cannot be overstated.

The main contributions presented in this paper include a) presenting the first corpus for abstractive Arabic text summarization, b) several neural models to perform abstractive news summarization for Arabic, and c) evaluation of the performance of these models. In addition to leveraging linguistic knowledge embodied in pretrained neural language models (using multilingual BERT, a transformer encoder trained on 104 Wikipedia languages including Arabic, and AraBERT, a monolingual BERT model trained specifically for Arabic), we also apply cross-lingual transfer to improve our results. We use summarization models based on multilingual neural language models (multilingual BERT and multilingual BART-50, a pre-trained sequence-to-sequence model for 50 languages including Arabic) that were originally fine-tuned to do summarization in another language, and further fine-tune them for the Arabic summarization task. We thus also leverage the knowledge of the original models concerning the summarization task they learned from resources in other languages.

The rest of the paper is structured as follows. Section 2 presents related work published on Arabic summarization. Section 3 describes the methodology and the experiments that we have done. Section 4 describes the results of automatic and manual evaluation. The last section concludes the paper.

2 Related Work

The work for Arabic summarization is limited. Most existing systems use the extractive approach. Lakhas (Douzidia and Lapalme, 2004) is considered the first extractive Arabic summarization system that was evaluated and compared with systems processing English input. The system produces a 10-word summary and translates it to English and then it is evaluated using the ROUGE measure

(Lin, 2004). Another Arabic text summarization approach based on fuzzy logic was proposed by Qassem et al. (2019). This model is based on a new noun extraction method and fuzzy logic. Yet another Arabic text summarization tool, SumSAT, (Lakhdar and Chéragui, 2019) adopts an extractive approach using a hybrid of three techniques: a) *contextual exploration* which allows access to the semantic content of a text, without the need for deep syntactic analysis; b) identification of *indicative expressions* offering the possibility of generating a summary in a general topic or a specific domain by selecting sentences that contain specific indicators, and c) the *graph method* which generates the summary by selecting the most representative phrases of the source text.

The evaluation process differs between these systems, as well as the datasets used for evaluation. Lakhas (the first extractive Arabic summarization system) was evaluated cross-lingually. It generates a summary, translates it to English, and then it evaluates the English summary using the ROUGE-N measure. For that aim 240 documents and their corresponding summaries were produced and used as a dataset. In the case of the SumSAT tool, performance was evaluated in terms of precision and recall of the discursive annotation generated by SumSAT and manual reference annotations (i.e. it was not summaries per se that were evaluated). For the evaluation, they constructed a dataset composed of 25 documents and their corresponding annotation. In contrast with the above, (Al Qassem et al., 2019) evaluated the summaries using ROUGE-N (N=1 and 2) metric and evaluated the summarizer using the Essex Arabic Summaries Corpus (EASC) (El-Haj et al., 2010), which contains 153 Arabic articles and 765 human-generated extractive summaries of those articles created using Mechanical Turk.

An RST-based[1] automatic summarization technique for Arabic texts was presented by (Maaloul et al., 2010) which was implemented through the ARSTRemue system. They created a corpus of Arabic texts from a newspaper website, and they claimed that the ARSTResume evaluation showed encouraging results based on 50 texts.

Some recently published work also aims to address abstractive Arabic text summarization. Azmi and Altmami (2018) proposed a four-phase abstractive summarizer for Arabic where the core of the system is an extractive summarizer. The four phases are topic segmentation, headline generation, extractive summarization, and sentence reduction. For evaluation, they conducted two experiments. The first is to evaluate the extractive summarizer. For that they used 32 sample documents from two popular Saudi newspapers. The second is to evaluate the abstractive summarizer. For that they used 150 documents from six different Arabic newspapers. In addition, two linguist experts judged the quality of the abstractive summaries.

Another system (Al-Maleh and Desouki, 2020) was trained to generate headlines based on the first paragraph of Arabic articles, a task that can be classified as a kind of abstractive summarization. The authors used a sequence-to-sequence model implementing the pointer-generator approach including a copy mechanism as presented in See et al. (2017). For training and evaluation, they crawled an Arabic data-set consisting of approximately 300 thousand *article headline : introductory paragraph* pairs.

An attempt at abstractive Arabic text summarization proper was presented in (Elmadani et al., 2020) applying multilingual-BERT-based (Devlin et al., 2019) models for both abstractive and extractive summarization using the models presented in (Liu and Lapata, 2019) trained and tested on the KALIMAT dataset (El-Haj and Koulali, 2013). A shortcoming of the research is, however, that the 20,291 article summaries in KALIMAT are machine-generated summaries output by the extractive Gen-Summ (=AQBTSS) algorithm (El-Haj et al., 2010). The train/test sets are thus neither human generated nor abstractive. Both studies evaluated the summaries using the ROUGE metric (Lin, 2004) .

3 Methodology

This paper reflects on a specific approach of abstractive text summarization applied to Arabic. In terms of model architecture, we focus on approaches based on now-ubiquitous large-scale pre-trained language models (LM), such as BERT (Devlin et al., 2019) and BART (Lewis et al., 2020), which obtained new state-of-the-art results in diverse natural language processing tasks, including text summarization. An important feature of BERT and BART is that both of them have a multilingual model available, M-BERT (Devlin et al., 2019) and M-BART-50 (Tang et al., 2020) that include Arabic among the languages supported. In addition, we

[1] Rhetorical Structure Theory

need a big enough training and evaluation dataset consisting of Arabic texts and their abstractive summaries. So, the first step we took was to compile a reliable Arabic abstractive news summary corpus.

3.1 Data Collection

While we could mention two recent papers attempting at something that could be categorized as abstractive summarization in Section 2, one of them dealt with headline generation instead of summarization proper, and the other used a machine-generated extractive summaries dataset for training and evaluation. The main bottleneck hindering progress in Arabic abstractive summarization is thus the lack of a sizable dataset. We first needed to overcome this problem. We needed to build an Arabic abstractive summarization corpus. A great source of such a resource could be the press, like in the case of the trend-setting CNN/Daily Mail dataset (See et al., 2017), as many news articles have a lead, a brief overview of the content spread out in the article, with details only supporting, but not altering the original message. The only problem is that news articles with reliable abstractive leads are difficult to find. It is quite often the case that the lead is just a copy of the first paragraph or contains clickbait content rather than containing a good abstractive summary.

Spending considerable effort on evaluating a wide range of sites from the Arabic versions of *CNN, BBC, France 24, DW, Sky News* to the most popular fully Arabic sites like *al-Mayadeen, al-Ālam, al-Ahrām, al-Akhbar,* and *Sada-elbalad,* we identified two Arabic news resources that could be the basis of a good Arabic abstractive news summaries dataset: the Arabic version of the *Deutsche Welle (DW)* news website[2], which seems to be the one containing the best abstractive summaries and the *Files* section of *Sada-elbalad.* The latter resource from Sada-elbalad later turned out to contain many problematic items containing several diverse topics only some of which were mentioned in the summary, we thus dropped this resource.

We downloaded Arabic Deutsche Welle resources from Common Crawl[3]. We only kept articles from the "Main/Top Stories" section, and filtered out all articles where either the main article text or the lead was too short or missing and items where the text is shorter than 4 times the length of the lead. The dataset that we used in the experiments consists of 21508 articles and their corresponding leads.

We performed data processing steps on the raw material (the collected articles) to be ready for subsequent processing. Data processing means a number of steps that naturally all differ significantly from one NLP task to another. While that is a sensitive process on its own, we also face another difficultly making it somewhat challenging to rely on the experiences of the already developed model. That is the peculiar nature of each language and that not much similar work has been done on Arabic, which has its own difficulties both as language and script.

We use Python since it is capable to handle the Arabic language. We also use NLTK platform since it is an appropriate tool for Arabic NLP and can be used for preprocessing text for text summarization task with Arabic. Based on our corpora we needed to perform text tokenization. Table1 displays the main characteristics of the corpora.

3.2 Experiments

The aim of the main task of our work is practically to fine-tune pre-trained language models for our task which is abstractive Arabic text summarization. For this aim, we fine-tuned multilingual BERT (having Arabic among the languages covered) for abstractive Arabic text summarization using our own corpus.

We also fine-tuned AraBERT (Antoun et al., 2020) for abstractive Arabic text summarization using the same corpus. AraBERT is the result of pre-training a BERT model specifically for the Arabic language.

In addition, we propose a cross-lingual-transfer-based approach to improve our results. Using pre-trained multilingual BERT, we fine-tuned multilingual BERT for abstractive Hungarian text summarization using the HVG[4] corpus (Yang et al., 2021) where the articles and corresponding leads were taken from a daily online newspaper. We further fine-tuned this model for abstractive Arabic text summarization using our own corpus.

We followed the same approach using English training data instead of Hungarian. We used the CNN/DailyMail summaries corpus containing over 300k unique news articles to first train an English summarization system fine-tuning multilin-

[2] https://www.dw.com/ar

[3] https://commoncrawl.org/

[4] https://hvg.hu/

	Articles	Leads
segments	21,508	
train	19,807	
test	1,701	
token #	6,929,974	2,867,754
type #	290,138	178,614
avg sent #	14.420	1.469
avg sent # (median)	13	2
avg token #	412.052	35.131
avg token # (median)	279	37
avg token # (mBERT)	848.821	73.559
avg token # (mBERT, median)	573	79
avg token # (araBERT)	481.219	39.631
avg token # (araBERT, median)	326	42
avg token # (mBART)	664.582	57.549
avg token # (mBART, median)	448	62

Table 1: Main characteristics of the corpus

gual BERT. Then we further fine-tuned this model for Arabic on our corpus.

We also fine-tuned the multilingual BART-50 model, which supports 50 languages including Arabic, using our own corpus. Following the approach mentioned above, we used a model fine-tuned from M-BART-50 for abstractive Russian text summarization using the Gazeta corpus (Gusev, 2020). We further fine-tuned this model for abstractive Arabic text summarization using our own corpus. Table 2 displays the ROUGE results of Hungarian and English m-BERT fine-tuning, and Russian m-BART-50 fine-tuning.

4 Results

Measuring the performance of a summarization system can be done through either automatic or manual evaluation. We evaluated our experiments using the ROUGE automatic metric and compared them to other abstractive Arabic summarization

Model	ROUGE-1	ROUGE-2	ROUGE-L
mBERT Hun	47.02	19.72	39.29
mBERT Eng	60.32	25.79	56.91
mBART Rus	32.1	14.2	25.7

Table 2: ROUGE recall results of Hungarian m-BERT, English m-BERT, and Russian m-BART fine-tuning

results. We also evaluated our results manually since the reliability of automatic metrics is often perceived as insufficient.

4.1 Automatic Evaluation

Automatic evaluation metrics are the most widely used tools in the overwhelming majority of the research papers on the subject of summarization. We have evaluated our experiments using ROUGE (Lin, 2004). ROUGE-1 and ROUGE-2 measure overlap of word uni-grams and bi-grams respectively. ROUGE-L measures overlap of the longest common sub-sequence between two texts. When comparing the performance of the models that we trained using our relatively small Arabic corpus, we found that using an abstractive summarization model based on multilingual BERT already fine-tuned for English on the CNN/DailyMail dataset as a starting point to train an Arabic summarization model leads to huge improvements in performance, as shown in Table 3.

Model	ROUGE-1	ROUGE-2	ROUGE-L
AraBERT	6.121	0.117	6.121
mBERT	5.134	0.186	5.134
mBERT+HUN	6.466	0.261	6.462
mBERT+ENG	**16.363**	**2.524**	**16.363**
mBART-50	6.817	0.382	6.809
mBART-50-rus	7.116	0.499	7.045

Table 3: ROUGE recall results of abstractive summarization

Automatic metrics are widely used to determine where a new system may rank against existing state-of-the-art systems. We thus compared our work with the latest Arabic research on abstractive text summarization, TRANS.ABS (Elmadani et al., 2020), the only one available, as shown in Table 4.

Note that although there is a significant difference between the measured performance, the numbers cannot be directly compared, because performance was measured on different test sets. Moreover, as it was mentioned in Section 2, TRANS.ABS was evaluated on KALIMAT, in

Model	ROUGE-1	ROUGE-2	ROUGE-L
TRANS.ABS*	6.93	1.78	6.88
mBERT+ENG	**12.61**	**2.11**	**12.61**

Table 4: Comparison of ROUGE F1 scores between existing abstractive Arabic summarization models. Result with * mark is taken from the corresponding paper.

which summaries are neither human-generated nor abstractive, so that corpus is not in fact suitable for the evaluation of abstractive summarization systems.

4.2 Manual Evaluation

In spite of the recent rapid progress in the development of summarization models, standard automatic evaluation metrics have not developed for nearly 20 years. In our experiments, ROUGE scores determine that our proposed method ranked significantly better than the existing systems, but the ROUGE scores did not reflect the real quality of the summaries generated. For the sake of a more accurate assessment, we decided to conduct a human evaluation. We manually evaluated the summaries generated by the different models. In order to achieve this, we created a web-based evaluation platform containing 100 random samples. For each of the 100 sample articles, the platform displays the following:

- Article text: the article text.
- Lead: the article corresponding lead.
- mBART-50: results generated from fine-tuning M-BART-50 with our corpus.
- mBART-50-ru-gazeta: results generated from fine-tuning the already fine-tuned M-BART-50 for Russian to Arabic.
- BERT multilingual cased trained from English model: results generated from fine-tuning the already fine-tuned M-BERT for English to Arabic.
- BERT multilingual cased trained from Hungarian model: results generated from fine-tuning the already fine-tuned M-BERT for Hungarian to Arabic.
- AraBERT: result generated from fine-tuning AraBERT with our corpus.
- BERT multilingual cased: result generated from fine-tuning multilingual BERT with our corpus.

The evaluation process was done by 3 human annotators, who are from different backgrounds and have different views. One (S) is from Syria, which is a Levant country, and Arabic is the annotator's mother tongue. The second (M) is from Morocco (Northwest Africa), where another dialect is used, and we can't say that Arabic is their spoken mother tongue. The third (H) is from Hungary, who is not a native speaker, but a professional translator. This variety of annotators, who all have different points of view and different approaches to the Arabic language, raises the evaluation standard and ensures more reliable results. Though the two native speakers are both proficient in Fusha, the minor regional stylistic differences and the difference in whether they rely on it as a primary or secondary language, give a different angle of evaluation. The Hungarian annotator, on the other hand, gives an outer, more "neutral" look to the annotation.

We conducted manual evaluation in two steps. The first step is "Ranking", we asked the annotators to evaluate the output of the models and assign marks to each summary from 1 to 6 as shown in Table 5.

Ranking
1 : BEST
2 : Very good
3 : Good
4 : Acceptable
5: Poor
6: Very poor

Table 5: Ranking scores for manual evaluation.

Given the results of the first step of evaluation, we chose the best model and asked the annotators for the second step of evaluation which is giving quality scores, in the range 1 to 5, concerning adequacy (to what extent the output covers most relevant information in the text) and fluency (Table 6).

The ranking results showed that the AraBERT model is the weakest model, while the model based on multilingual BERT first trained for English summarization and then fine-tuned for Arabic (m-BERT English) is the best-performing model.

The manual evaluation showed that the six mod-

Adequacy	Fluency
1 : none	1: incomprehensible
2 : little meaning	2: dis-fluent Arabic
3 : much meaning	3: non-native Arabic
4 : most meaning	4: good Arabic
5 : all meaning	5: flawless Arabic

Table 6: Adequacy and Fluency scores for manual evaluation.

els differ considerably, though in several areas they are difficult to compare. Output from the m-BERT English model usually comes very close to the original lead. Clarity and language proficiency is rarely a problem.

The m-BERT-based model first fine-tuned for Hungarian (m-BERT Hungarian) also generates good summaries, but usually in a very different way. The wording, structural order, and grammatical tools have a tendency to differ, but in most cases, the meaning does not change. It is usual for this model to generate somewhat (about 10%) longer summaries, but added content is usually explanatory rather than just simple text addition. In other words, these additions give depth to the summary and structural coherence. However, comparison is difficult due to the significantly different expressions used. In other words, while the English-trained BERT model almost recreates the original lead, the Hungarian-trained one formulates the content in a different way.

Summaries generated by the model simply fine-tuned from multilingual BERT without pre-training on summaries in another language correlate with those generated by the the English-trained model, but they contain significantly more grammatical and contextual errors. Sometimes the message is just the opposite of that of the original article, sometimes the syntax falls apart. Yet there is a good number of promising summaries. It seems like a promising model still in development. It seems unfinished.

The AraBert-based model is by far the weakest. It is clearly insufficient for practical usage. This model has a notorious tendency of distorting or reversing the meaning of the text, coming up with disturbingly wrong interpretations. There is a very high number of huge, sometimes hilarious grammatical mistakes not present in the output of any other model. Most problematic, however, is that this model generates by far the longest summaries. Often the size is double of that the original lead or the output of the first model, yet this lengthy text does not add anything relevant to the summary. It simply bloats the summary, but does not add content. It seems like randomly selected and poorly sewn together sentences from the original text itself, but with great alterations of the meaning.

The models based on multilingual BART-50 and the m-BART-50-based model first fine-tuned for Russian have almost equally good results to the m-BERT English model, with remarkable text quality and fluency.

There seems to be little chance for improvement for the AraBert-based model, unlike the others, which are very promising. For some of the summaries it cannot be determined whether they were written by a human editor or are machine-generated. See Figure 1,2. Table 7 shows the Kappa(Cohen, 1960) values of inter-annotator agreement. We used 4 metrics to measure the inter-annotator agreement: Fleiss's Kappa (Fleiss), Krippendorff's alpha coefficient (Krippen), Scott's pi (Scott), Average Pairwise Cohen's Kappa (Cohen). The values of the inter-annotator agreement for the m-BERT English model are substantial.

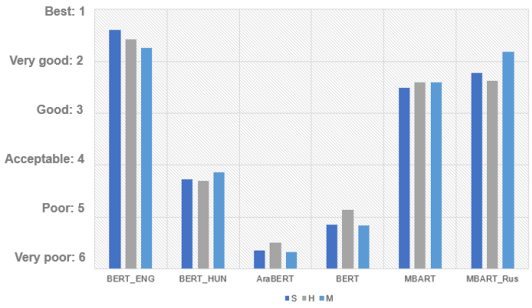

Figure 1: Ranking results

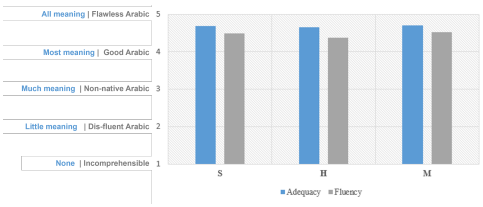

Figure 2: M-BERT_English Adequacy and Fluency manual evaluation results

5 Conclusion

In this paper, we introduced the first corpus for abstractive Arabic text summarization, which we compiled with our own effort. Based on this corpus,

	Ranking						AD	FL
	mBERT-E	mBERT-H	mBERT	AraBERT	mBART	mBART-R	mBERT-Eng	
Fleiss	0.613	0.285	0.312	0.310	-0.016	-0.009	0.228	0.175
Krippen	0.613	0.273	0.301	0.303	-0.015	-0.043	0.229	0.177
Scott	0.612	0.273	0.298	0.304	-0.019	-0.047	0.226	0.174
Cohen	0.613	0.298	0.317	0.308	-0.016	-0.005	0.232	0.174

Table 7: Evaluation of inter-annotator agreement

we fine-tuned multilingual-BERT and multilingual-BART-based models for Arabic abstractive summarization. We also proposed a cross-lingual-knowledge-transfer-based approach. We applied this approach to improve summarization quality, further fine-tuning models first fine-tuned from multilingual BERT for Hungarian or English summarization to generate Arabic summaries, and applying the same training scenario to Russian using an M-BART-50-based model. The results of the ROUGE metric and manual evaluation showed that the proposed approach led to significant improvements in performance and achieved state-of-the-art results. In the future, we would like to extend our corpus and perform experiments with other models such as the PEGASUS model.

Acknowledgments

The authors of the paper wish to express the deepest gratitude to Professor Gábor Prószéky for his unrelenting support and to Dr. Dániel Sógor and Youssef Messaoudi for their great work in the manual evaluation process. This research was implemented with partial support provided by the National Excellence Programme 2018-1.2.1-NKP-00008:*"Exploring the Mathematical Foundations of Artificial Intelligence"*.

References

Molham Al-Maleh and Said Desouki. 2020. Arabic text summarization using deep learning approach. *Journal of Big Data*, 7:1–17.

Lamees Al Qassem, Di Wang, Hassan Barada, Ahmad Al-Rubaie, and Nawaf Almoosa. 2019. Automatic Arabic text summarization based on fuzzy logic. In *Proceedings of the 3rd International Conference on Natural Language and Speech Processing*, pages 42–48.

Wissam Antoun, Fady Baly, and Hazem Hajj. 2020. AraBERT: Transformer-based model for Arabic language understanding. In *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection*, pages 9–15, Marseille, France. European Language Resource Association.

Aqil M. Azmi and Nouf I. Altmami. 2018. An abstractive arabic text summarizer with user controlled granularity. *Information Processing and Management*, 54(6):903–921.

J. Cohen. 1960. A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, 20(1):37.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Fouad Soufiane Douzidia and Guy Lapalme. 2004. Lakhas, an Arabic summarization system. *Proceedings of DUC2004*.

Mahmoud El-Haj and R. Koulali. 2013. KALIMAT a multipurpose Arabic corpus. In *Second Workshop on Arabic Corpus Linguistics (WACL-2)*, pages 22–25.

Mahmoud El-Haj, Udo Kruschwitz, and C. Fox. 2010. Using Mechanical Turk to create a corpus of Arabic summaries. In *Language Resources (LRs) and Human Language Technologies (HLT) for Semitic Languages workshop in conjunction with the 7th International Language Resources and Evaluation Conference (LREC 2010)*.

Khalid N. Elmadani, Mukhtar Elgezouli, and Anas Showk. 2020. BERT fine-tuning for Arabic text summarization. *ArXiv*, abs/2004.14135.

Ilya Gusev. 2020. Dataset for automatic summarization of russian news. *AINL 2020. Communications in Computer and Information Science, vol 1292. Springer, Cham (2020)*.

Said Moulay Lakhdar and Mohamed Amine Chéragui. 2019. Building an extractive Arabic text summarization using a hybrid approach. In *International Con-*

ference on Arabic Language Processing, pages 135–148. Springer.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Yang Liu and Mirella Lapata. 2019. 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)*, pages 3730–3740, Hong Kong, China. Association for Computational Linguistics.

Mohamed Hedi Maaloul, Iskandar Keskes, Lamia Belguith, and Philippe Blache. 2010. Automatic summarization of Arabic texts based on RST technique. In *Proceedings of the 12th International Conference on Enterprise Information Systems - Artificial Intelligence and Decision Support Systems*, pages 434–437.

Ramesh Nallapati, Bowen Zhou, and Mingbo Ma. 2016. Classify or select: Neural architectures for extractive document summarization. *arXiv preprint arXiv:1611.04244*.

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. Ranking sentences for extractive summarization with reinforcement learning. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1747–1759, New Orleans, Louisiana. Association for Computational Linguistics.

Romain Paulus, Caiming Xiong, and Richard Socher. 2017. A deep reinforced model for abstractive summarization. *CoRR*, abs/1705.04304.

Lamees Al Qassem, Di Wang, Hassan Barada, Ahmad Al-Rubaie, and Nawaf Almoosa. 2019. Automatic Arabic text summarization based on fuzzy logic. In *Proceedings of the 3rd International Conference on Natural Language and Speech Processing*, pages 42–48, Trento, Italy. Association for Computational Linguistics.

Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389, Lisbon, Portugal. Association for Computational Linguistics.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. 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)*, pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

Yuqing Tang, Chau Tran, Xian Li, Peng-Jen Chen, Naman Goyal, Vishrav Chaudhary, Jiatao Gu, and Angela Fan. 2020. Multilingual Translation with Extensible Multilingual Pretraining and Finetuning. *arXiv e-prints*, page arXiv:2008.00401.

Zijian Győző Yang, Ádám Agócs, Gábor Kusper, and Tamás Váradi. 2021. Abstractive text summarization for Hungarian. In *The 1st Conference on Information Technology and Data Science*, pages 299–316.

Xingxing Zhang, Mirella Lapata, Furu Wei, and Ming Zhou. 2018. Neural latent extractive document summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 779–784, Brussels, Belgium. Association for Computational Linguistics.