Natural Language Engineering (2023), **29**, pp. 1275–1304
doi:10.1017/S1351324922000195

CAMBRIDGE UNIVERSITY PRESS

ARTICLE

Turkish abstractive text summarization using pretrained sequence-to-sequence models

Batuhan Baykara* and Tunga Güngör

Department of Computer Engineering, Boğaziçi University, Bebek, 34342 Istanbul, Turkey
*Corresponding author. E-mail: batuhan.baykara@boun.edu.tr

(Received 11 October 2021; revised 16 April 2022; accepted 18 April 2022; first published online 13 May 2022)

Abstract

The tremendous amount of increase in the number of documents available on the Web has turned finding the relevant piece of information into a challenging, tedious, and time-consuming activity. Accordingly, automatic text summarization has become an important field of study by gaining significant attention from the researchers. Lately, with the advances in deep learning, neural abstractive text summarization with sequence-to-sequence (Seq2Seq) models has gained popularity. There have been many improvements in these models such as the use of pretrained language models (e.g., GPT, BERT, and XLM) and pretrained Seq2Seq models (e.g., BART and T5). These improvements have addressed certain shortcomings in neural summarization and have improved upon challenges such as saliency, fluency, and semantics which enable generating higher quality summaries. Unfortunately, these research attempts were mostly limited to the English language. Monolingual BERT models and multilingual pretrained Seq2Seq models have been released recently providing the opportunity to utilize such state-of-the-art models in low-resource languages such as Turkish. In this study, we make use of pretrained Seq2Seq models and obtain state-of-the-art results on the two large-scale Turkish datasets, TR-News and MLSum, for the text summarization task. Then, we utilize the title information in the datasets and establish hard baselines for the title generation task on both datasets. We show that the input to the models has a substantial amount of importance for the success of such tasks. Additionally, we provide extensive analysis of the models including cross-dataset evaluations, various text generation options, and the effect of preprocessing in ROUGE evaluations for Turkish. It is shown that the monolingual BERT models outperform the multilingual BERT models on all tasks across all the datasets. Lastly, qualitative evaluations of the generated summaries and titles of the models are provided.

Keywords: Abstractive text summarization; News title generation; Pretrained sequence-to-sequence models

1. Introduction

With the emergence of the Web, there has been an exponential increase in the number of documents made available online from sources such as websites, news, blogs, books, scientific papers, and social media. In parallel to this, it has become increasingly difficult for users to find the information they are interested in due to repetitive and irrelevant content. Moreover, the time and effort that are required to comprehend all these sources are immense. There is a need to automatically digest and extract the essence of all this information since it is impractical for humans to comprehend this vast amount of information through manual efforts. In this regard, text summarization has become an inevitable necessity and a very popular field of study in the past few decades.

Text summarization aims at automatically generating a concise piece of text from a long document, which is capable of portraying the most important piece of information in a fluent and salient way (Luhn 1958; Edmundson 1969). There are two main approaches in automatic text summarization: extractive text summarization (Mihalcea and Tarau 2004; Nallapati, Zhai, and Zhou 2017) and abstractive text summarization (Rush, Chopra, and Weston 2015; See, Liu, and Manning 2017; Zhang *et al*. 2020). Extractive text summarization produces summaries by selecting the most relevant sentences or phrases from the input text without reflecting any changes. Abstractive text summarization, on the other hand, is a more challenging task where the aim is to generate a human like summary through making use of complex natural language understanding and generation capabilities.

Abstractive text summarization has gained much more popularity after the advances in deep learning. Recently, sequence-to-sequence (Seq2Seq) models with encoder-decoder architecture have dominated the field. The underlying components of encoder-decoder networks have shifted from LSTM-based (Hochreiter and Schmidhuber 1997) to transformer-based (Vaswani *et al*. 2017) models. Language model pretraining (Radford *et al*. 2018; Devlin *et al*. 2019) has advanced the state-of-the-art, especially for natural language understanding, in numerous natural language processing (NLP) tasks. These pretrained language models have also been adopted in abstractive text summarization (Liu and Lapata 2019). Later, studies leveraged the pretraining for Seq2Seq models (Dong *et al*. 2019; Song *et al*. 2019; Rothe, Narayan, and Severyn 2020; Lewis *et al*. 2020; Raffel *et al*. 2020) to further improve upon the language generation tasks. Accordingly, pretraining Seq2Seq models especially on large-scale datasets have shown to perform very well, reaching state-of-the-art results in neural abstractive summarization (Zhang *et al*. 2020; Qi *et al*. 2020).

Unfortunately, all these research attempts have been mostly limited to the English language. Additionally, pretraining such models requires vast amount of data and computational power which are factors that limit research. However, multilingual versions of the BERT (Devlin *et al*. 2019) model and two multilingual pretrained Seq2Seq models (mT5 Xue *et al*. 2021 and mBART Liu *et al*. 2020) have been released recently. This has given rise to many possibilities in various research areas for low-resourced languages. Moreover, many monolingual BERT models in various languages have been pretrained by the community including BERTurk (Schweter 2020), a monolingual Turkish BERT model.

Text summarization studies in Turkish are mostly based on extractive approaches. There are very few studies that try to tackle the abstractive summarization task in Turkish (Scialom *et al*. 2020; Baykara and Güngör 2022). None of these works has made use of pretrained Seq2Seq models which have shown to reach state-of-the-art results for English. Additionally, title generation is also considered as a text summarization task since the main objective is to output a condensed summary in the form of a title (Rush *et al*. 2015). However, the number of title generation studies in Turkish is very limited (Karakoç and Yılmaz 2019). There are currently two large-scale datasets, TR-News (Baykara and Güngör 2022) and MLSum (Scialom *et al*. 2020), which are suitable for Turkish abstractive text summarization. In this study, we aim to leverage these pretrained models for the abstractive text summarization and title generation tasks on the TR-News and MLSum datasets and provide detailed analyses of the obtained results.

We address the following research questions in this paper:

- RQ1: How do pretrained sequence-to-sequence models perform on Turkish abstractive text summarization and title generation tasks?
- RQ2: Does the monolingual BERT model obtain better results than the multilingual BERT model on the BERT2BERT model architecture?
- RQ3: Does combining datasets with similar characteristics improve model performance in abstractive text summarization and title generation?

https://doi.org/10.1017/S1351324922000195 Published online by Cambridge University Press

- RQ4: How do models trained on one dataset perform across other datasets that have similar characteristics?
- RQ5: How much does the input to a title generation model impact the model performance?

Inline with the research questions, the contributions in this paper are as follows:[a]

- We show that pretrained sequence-to-sequence models reach state-of-the-art on the TR-News and MLSum datasets for summary generation and title generation tasks.
- We conduct the first study that utilizes the titles of both datasets, and we provide comprehensive and strong baselines for the title generation task.
- We show that monolingual BERTurk models outperform the multilingual BERT models on BERT2BERT architecture.
- We observe that combining both datasets yields better models for both text summarization and title generation tasks.
- We conduct cross-dataset evaluations for both tasks and show that the models trained on TR-News are more robust compared to those trained on MLSum.
- We measure the efficacy of providing different inputs (LEAD sentences vs. abstract) to a Seq2Seq model for title generation task and demonstrate that the abstract proves to be a better option compared to the LEAD sentences.
- We show how much preprocessing affects the ROUGE calculations, which is especially important for agglutinative languages like Turkish.

The rest of the paper is organized as follows. In Section 2, we give an overview of the recent literature on abstractive text summarization. This is followed by Sections 3 and 4 where, respectively, the models and the datasets used in this study are presented. Section 5 discusses the experimental setup, an analysis of the tokenization methods used in the models, and the novelty measurements for both text summarization and title generation tasks. The quantitative and qualitative results of the experiments are presented in Section 6. We conclude the paper in Section 7.

2. Related work

2.1 Pretrained sequence-to-sequence models

In recent years, transfer learning in NLP has proven to be very effective and has enabled state-of-the-art results in a large variety of tasks. The concept of pretraining a language model that is capable of learning task-agnostic knowledge through various pretraining objectives and then transferring this knowledge to downstream tasks has been especially successful in natural language understanding (Radford *et al.* 2018; Devlin *et al.* 2019; Yang *et al.* 2019). However, tasks that require both natural language understanding and natural language generation such as machine translation and text summarization could not benefit from these pretrained encoder models as much, leading to pretrained sequence-to-sequence models.

Song *et al.* (2019) proposed MASS, a masked Seq2Seq generation model, that is able to reproduce part of a sentence when the remaining parts are provided. UniLM (Dong *et al.* 2019) employed simultaneous training on three types of language modeling objectives: unidirectional, bidirectional, and sequence-to-sequence. In BART, Lewis *et al.* (2020) followed various denoising objectives to first corrupt an input text and then reconstruct it using an autoencoder. T5 (Raffel *et al.* 2020) introduced a generalized text-to-text framework capable of handling a variety of NLP tasks using solely text as its input and output, and is pretrained on various supervised

[a] All the available code has been made publicly available at https://github.com/batubayk/enc_dec_sum.

https://doi.org/10.1017/S1351324922000195 Published online by Cambridge University Press

and unsupervised objectives including summarization. Lastly, the multilingual variations of T5 and BART, respectively, mT5 (Xue *et al*. 2021) and mBART (Liu *et al*. 2020), were released.

2.2 Abstractive text summarization

Abstractive text summarization is dominantly framed as a sequence-to-sequence problem and encoder-decoder networks are frequently used to tackle this problem. Rush *et al*. (2015) were one of the first studies to apply an encoder-decoder architecture using a neural network language model (NNLM) to the title generation task as part of the abstractive summarization problem. Then, Chopra, Auli, and Rush (2016) replaced the NNLM with recurrent neural networks (RNNs). Nallapati *et al*. (2016) introduced several novel models including a bidirectional LSTM-based encoder-decoder with attention mechanism, a model with a feature rich encoder, a switching pointer-generator model, and a hierarchical encoder-decoder that is capable of capturing the document structure. Additionally, converting the CNN/Daily Mail dataset (Hermann *et al*. 2015) into a format for text summarization was also amongst their contributions. The pointer-generator model was enhanced allowing it to copy words from the source document, and a coverage mechanism was added to tackle the word repetition problem (See *et al*. 2017). Later, various reinforcement learning models were applied to neural abstractive summarization (Çelikyılmaz *et al*. 2018; Paulus, Xiong, and Socher 2018). Convolutional neural networks (CNNs) were used jointly with topic aware embeddings on the XSum dataset to better capture the theme of the documents (Narayan, Cohen, and Lapata 2018a). The pretrained language model BERT was adopted as the encoder component for better language understanding capability (Liu and Lapata 2019).

Recently, the pretrained Seq2Seq models have shown to perform very well for neural abstractive summarization (Lewis *et al*. 2020; Raffel *et al*. 2020). PEGASUS (Zhang *et al*. 2020) was specifically pretrained for the abstractive text summarization task and made use of masking whole sentences from a document and generating these gap sentences as the pretraining objective. ProphetNet (Qi *et al*. 2020) introduced a novel self-supervised objective named as future *n*-gram prediction and the n-stream self-attention mechanism. Unlike traditional Seq2Seq models which optimize one-step ahead prediction, it optimizes n-steps ahead predicting the next n tokens simultaneously based on previous context tokens at each time step.

2.3 Turkish text summarization

The research in Turkish text summarization is mostly based on extractive approaches where more traditional methods are utilized. In an early work, a rule-based system which aims to summarize news articles through various heuristics has been proposed (Altan 2004). For instance, more importance is given to the sentences that contain positive sentiments or that are at the introduction or conclusion parts of the input text. Other studies made use of features that are commonly used in extractive text summarization such as term frequency, title similarity, key phrases, and sentence position and centrality to select the most relevant sentences (Çığır, Kutlu, and Çiçekli 2009; Kartal and Kutlu 2020). Özsoy, Çiçekli, and Alpaslan (2010) proposed variations to the commonly applied latent semantic analysis (LSA) such as finding the main topics of the text and then selecting the sentences that have the highest scores amongst those topics. Query-biased summarization was studied in the Web information retrieval domain to further improve snippet quality by utilizing the document structure (Pembe and Güngör 2008). Güran, Bayazit, and Bekar (2011) made use of non-negative matrix factorization and applied various preprocessing methods such as detecting consecutive words, removing stopwords, and stemming. Later, a hybrid extractive summarization system was proposed which uses semantic features extracted from Wikipedia in conjunction with the commonly used structural features (Güran, Bayazit, and Gürbüz 2013).

The number of studies on Turkish abstractive text summarization is very limited as well as the applications of pretrained Seq2Seq models on Turkish text summarization and title generation

https://doi.org/10.1017/S1351324922000195 Published online by Cambridge University Press

tasks. Scialom *et al*. (2020) evaluated the recent Seq2Seq models (pointer generator See *et al*. 2017 and UniLM Dong *et al*. 2019) on the MLSum dataset that they have released, which consists of five different languages including Turkish. Karakoç and Yılmaz (2019) employed a plain LSTM-based encoder-decoder network for the title generation task. Baykara and Güngör (2022) proposed several morphological tokenization methods and evaluated them using the pointer-generator (See *et al*. 2017) model and also compared the results to BERT-based models (Liu and Lapata 2019) on the TR-News dataset which they have released.

3. Models

In recent years, pretraining a sequence-to-sequence model and finetuning it on downstream tasks such as machine translation and text summarization has shown to be very effective yielding state-of-the-art results in English. Until very recently, these models were mostly limited to the English language and it was not possible to assess the performance of such models in other languages. Pretraining these models require vast amount of data, computational resources, and budget, so obtaining these models for other languages is highly challenging. These limitations have been addressed in the recent multilingual pretrained sequence-to-sequence models mBART (Liu *et al*. 2020) and mT5 (Xue *et al*. 2021). In this work, we utilize these two pretrained multilingual sequence-to-sequence models and also warm-start sequence-to-sequence (BERT2BERT) models from pretrained BERT models.

3.1 BERT2BERT

BERT (Devlin *et al*. 2019) is a bidirectional transformer network pretrained on a large corpus with two pretraining objectives: masked language modeling and next sentence prediction. It closely follows the original transformer network (Vaswani *et al*. 2017) with the major improvement being the bidirectional self-attention mechanism. The authors have released several multilingual pretrained models that support a wide variety of languages including Turkish. In addition to the multilingual models, monolingual models have been pretrained by the community (Virtanen *et al*. 2019; Polignano *et al*. 2019; Kuratov and Arkhipov 2019; Chan, Schweter, and Möller 2020; Schweter 2020). Tokenization is an important aspect for these models since the input tokens are directly determined by the tokenization method and accordingly might impact the models' performance (Bostrom and Durrett 2020; Zhang *et al*. 2020). Most of the released models follow the original BERT model and were pretrained using the WordPiece (Wu *et al*. 2016) tokenization method.

Unlike sequence-to-sequence models which are composed of two parts, an encoder and a decoder, BERT works as an encoder-only model. Figure 1 shows a high-level view of a sequence-to-sequence transformer encoder-decoder model. The encoder transformer layers usually contain bidirectional connections which closely resemble the BERT model, whereas the decoder layers contain unidirectional (left to right) connections. Although BERT is an encoder-only model, it is possible to utilize pretrained checkpoints so that a sequence-to-sequence model can be constructed by initializing both the encoder and the decoder parts with pretrained model checkpoints (Rothe *et al*. 2020). This procedure is known as warm-starting an encoder-decoder model. In order to achieve this objective with BERT, (1) a randomly initialized cross attention layer is added in between the self-attention layers and the feed-forward layers in the decoder layers, (2) BERT's bidirectional self-attention layers in the decoder are changed to unidirectional self-attention layers, and (3) a language model layer is added on top of the decoder component to define a conditional probability distribution while generating outputs. Consequently, the pretrained weights are directly transferred to the constructed encoder-decoder model with the only exception being the additional cross attention layers which are randomly initialized.

https://doi.org/10.1017/S1351324922000195 Published online by Cambridge University Press

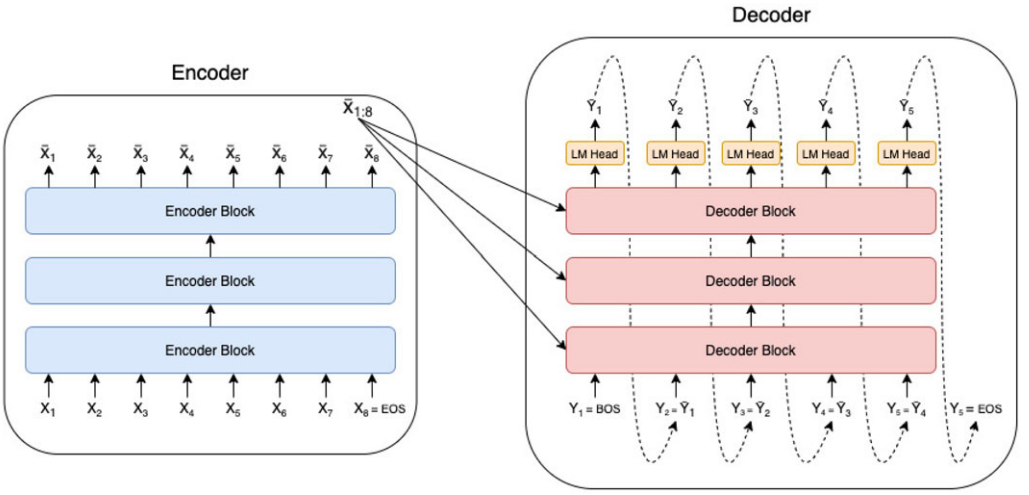

Figure 1. A high-level transformer-based encoder-decoder network.

In this work, we use both the multilingual BERT model (Devlin *et al.* 2019) and the monolingual Turkish BERT model called BERTurk (Schweter 2020), and their cased and uncased variations to warm-start pretrained sequence-to-sequence models. In the experiments part, the BERT2BERT models will be referred with their BERT model names (e.g., uncased multilingual BERT (mBERT-uncased) or uncased BERTurk (BERTurk-uncased)).

3.2 mBART

mBART (Multilingual Bidirectional and Auto-Regressive Transformers) (Liu *et al.* 2020) is the multilingual variation of the BART model (Lewis *et al.* 2020). BART is a pretrained encoder-decoder transformer network mostly suited to sequence-to-sequence tasks. The model is composed of a bidirectional encoder which closely resembles the BERT model (Devlin *et al.* 2019) and an autoregressive decoder that takes its roots from the GPT (Generative Pretrained Transformer) model (Radford *et al.* 2018). The BERT model is known to be more effective in language understanding tasks, whereas GPT-based models perform better in language generation tasks. Therefore, the BART model combines the strong aspects of both BERT and GPT-based models. Two different BART models have been released: base and large where the number of transformer layers for these models are 6 and 12, respectively. On the other hand, only one model size for mBART has been released which has 12 transformer layers with a model dimension of 1024 on 16 heads.

Similar to other pretrained models, BART makes use of several pretraining objectives and the main objective is to use denoising elements to corrupt the input and expect the model to reconstruct the original input. Hence, in principal it works as a denoising autoencoder. The noising methods on the input include token masking, token deletion, text infilling, sentence permutation, and document rotation which are displayed in Figure 2. The token masking operation randomly chooses tokens in the text and masks these tokens, whereas the token deletion operation deletes them. The text infilling method is similar to token masking but instead of choosing a single token, a span of tokens is chosen and masked where the span length is obtained from a Poisson distribution ($\lambda = 3$). The sentence permutation operation changes the order of the sentences and the document rotation operation shifts the entire text based on a randomly chosen token. The authors decided on a combination of text infilling and sentence permutation methods for the pretraining objective after completing extensive evaluations. The same approaches are also applied to the mBART model.

https://doi.org/10.1017/S1351324922000195 Published online by Cambridge University Press

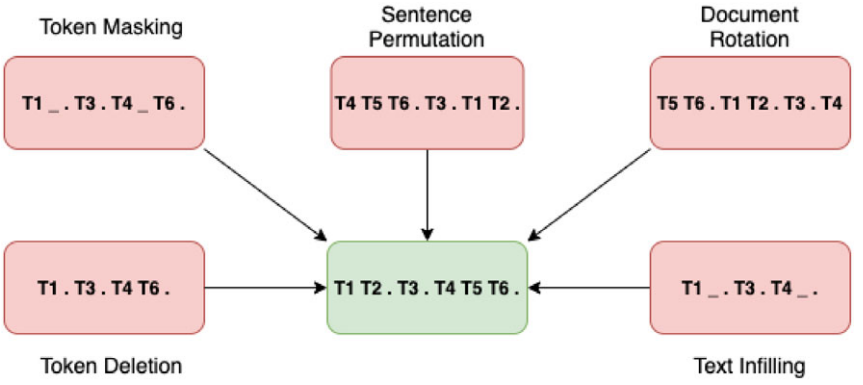

Figure 2. A number of noising methods experimented in the BART model. T1-T6 denote tokens. The box that the arrows point to shows the denoised text.

The BART model was pretrained on a combination of several resources such as books, news, web text, and stories following the work of Liu *et al.* (2019), whereas a subset of the Common Crawl (CC) corpus containing 25 languages (Wenzek *et al.* 2020) was used to pretrain the mBART model. Byte-pair encoding (BPE) (Sennrich, Haddow, and Birch 2016) method was used in the tokenization process of the BART model, and SentencePiece (Kudo 2018) tokenization was utilized in the pretraining of the mBART model. Two mBART models have been released: mbart-large-cc25 and mbart-large-50 where the models have been trained on 25 and 50 languages, respectively. In this work, we use the mbart-large-cc25 model and refer to it as the mBART model.

3.3 mT5

mT5 (Multilingual Text-to-Text Transfer Transformer) (Xue *et al.* 2021) is the multilingual variant of the T5 model (Raffel *et al.* 2020) and does not incorporate any major changes in terms of the model architecture. The T5 model is a sequence-to-sequence encoder-decoder network which closely follows the originally proposed transformer architecture (Vaswani *et al.* 2017) with some minor modifications. The main idea behind the T5 model is to approach each text related task as a text-to-text problem where the system receives a text sequence as an input and outputs another text sequence. This approach enables the system to use the same model and objective (teacher-forced maximum likelihood) for every downstream task. In that sense, T5 is an NLP framework capable of handling various tasks such as text summarization, question answering, text classification, and even tasks with continuous outputs such as semantic textual similarity under one unified framework. Figure 3 depicts the overall mT5/T5 models as a unified framework of various downstream tasks.

T5 makes use of several pretraining objectives to provide the model with generic capabilities which can be leveraged in downstream tasks. These include unsupervised objectives such as prefix language modeling, masked language modeling, and deshuffling along with several supervised objectives such as machine translation, text summarization, and text classification. As seen in Figure 3, each required task needs to be addressed with its corresponding prefix in the input sequence. For instance, the text summarization task requires the "summarize:" prefix, whereas the machine translation task requires the "translate English to Turkish:" prefix. The same approaches are also applied to mT5.

The pretraining of T5 was performed on the Colossal Clean Crawled Corpus (C4) (Raffel *et al.* 2020) which is only suited to the English language, whereas another dataset called mC4 was derived from Common Crawl[b] for pretraining the mT5 model on 101 different languages (Xue

[b] https://commoncrawl.org/.

https://doi.org/10.1017/S1351324922000195 Published online by Cambridge University Press

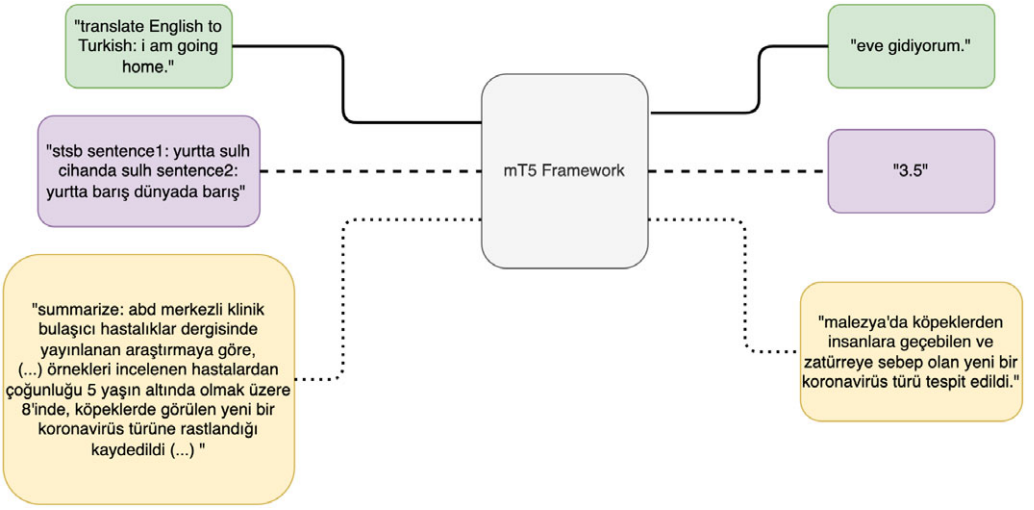

Figure 3. Various downstream tasks such as machine translation, semantic textual similarity, and text summarization on mT5 framework shown with examples in Turkish.

et al. 2021). The SentencePiece (Kudo 2018) algorithm is used in mT5 to cover a large multilingual vocabulary size of 250,000 which is several magnitudes higher compared to the original T5 vocabulary size of 32,000. The authors released several model sizes (small, base, large, xl, and xxl) for both T5 and mT5, and compared the model performances for the English language on the SQuAD reading comprehension benchmark (Rajpurkar *et al.* 2016) after finetuning the models to determine possible performance degradations. It was shown that mT5 falls behind the T5 model on all model sizes where the gap being smaller for larger models. Moreover, the mT5-xxl model reaches the state-of-the-art results in tasks such as paraphrase identification, natural language inference, and question answering on the XTREME multilingual benchmark (Hu *et al.* 2020) when compared to other multilingual pretrained models such as multilingual BERT (Devlin *et al.* 2019) and XLM-R (Conneau *et al.* 2020). In this work, we use the mT5-base model due to computational restrictions that the larger models bring and refer to it as mT5 in our experiments.

4. Datasets

Most of the research in text summarization and in NLP in general is based on the English language. Accordingly, the number of resources available for other languages such as Turkish is very limited. The datasets used in Turkish text summarization studies were limited in the number of training samples ranging just between 50 and 120 (Çığır *et al.* 2009; Özsoy *et al.* 2010). Due to this restriction, as stated in Section 2.3, the studies were overwhelmingly in an extractive manner. However, recent dataset contributions such as MLSum (Scialom *et al.* 2020) and TR-News (Baykara and Güngör 2022) have made it possible to work on abstractive text summarization with large-scale datasets for the Turkish language. MLSum is intended as a multilingual text summarization dataset covering five languages: French, German, Spanish, Turkish, and Russian. The TR-News dataset was compiled from three different news websites: Cumhuriyet,[c] NTV,[d] and HaberTürk,[e] whereas the Turkish subset of MLSum was obtained from a single website, İnternet

[c] https://cumhuriyet.com/.
[d] https://www.ntv.com.tr/.
[e] https://www.haberturk.com/.

https://doi.org/10.1017/S1351324922000195 Published online by Cambridge University Press

Table 1. Comparison of summarization datasets with respect to sizes of training, validation, and test sets, and average content, abstract, and title lengths (in terms of words and sentences)

		Content		Abstract		Title
Datasets	Num docs (train/val/test)	words	sentences	words	sentences	words
TR-News	277,573/14,610/15,379	286.18	15.72	25.05	1.48	6.53
MLSum (TR)	249,277/11,565/12,775	309.08	17.44	22.87	1.55	6.46
Combined-TR	526,850/26,175/28,154	296.97	16.53	24.02	1.51	6.50
CNN/Daily Mail	287,113/13,368/11,490	785.94	37.82	55.06	3.70	–
XSum	204,045/11,332/11,334	429.47	18.38	23.19	1.00	–

Table 2. Comparison of summarization datasets with respect to vocabulary size and type-token ratio of content, abstract, title, and overall

	Vocabulary size				Type-token ratio			
Datasets	content	abstract	title	overall	content	abstract	title	overall
TR-News	1,186,230	267,275	133,597	1,219,194	0.0135	0.0394	0.0665	0.0125
MLSum (TR)	1,109,917	228,511	109,628	1,143,534	0.0131	0.0365	0.0620	0.0123
Combined-TR	1,679,060	359,809	177,865	1,730,074	0.0097	0.0258	0.0471	0.0091
CNN/Daily Mail	869,792	240,663	–	893,985	0.0035	0.0140	–	0.0034
XSum	436,635	83,626	–	441,566	0.0045	0.0160	–	0.0043

Haber.[f] Both datasets cover news articles from a wide range of topics. In this work, we use the TR-News dataset and the Turkish subset of MLSum which we refer to as MLSum (TR).

News-based datasets compiled for text summarization comprise of news articles and one or more reference summary for each article. The reference summary is normally constructed by human evaluators. However, for large-scale datasets this is a very tedious work. Consequently, the reference summaries of these datasets are formed of the abstract part (highlight field) of the news articles (Nallapati *et al*. 2016; Scialom *et al*. 2020). In this work, in addition to the news article and the abstract, we also leverage the titles in a separate title generation task which is considered as another type of summarization task (Rush *et al*. 2015; Qi *et al*. 2020).

Table 1 shows the Turkish datasets used in this study. We also provide in the second part of the table two commonly used English summarization datasets, CNN/Daily Mail (Nallapati *et al*. 2016) and XSum (Narayan *et al*. 2018a), for comparison. As can be seen, TR-News and MLSum (TR) are similar in terms of the number of documents. Another important aspect for summarization tasks is the lengths of content, abstract, and title in number of words and sentences. These two datasets are similar to the XSum dataset with respect to the average number of sentences in the abstracts, which only contains one sentence per summary. This is partly due to the agglutinative nature of Turkish where the same information can be expressed with fewer words when compared to other languages such as English. Given the similar characteristics of TR-News and MLSum (TR), we combined these two datasets to see whether increasing the number of training samples would lead to a possible increase in model performances. We refer to the combined dataset as the Combined-TR.

The total number of distinct words (vocabulary size) and the type-token ratios for each dataset are given in Table 2. Type-token ratio (TTR) is calculated by dividing the vocabulary size to the

[f] https://www.internethaber.com/.

https://doi.org/10.1017/S1351324922000195 Published online by Cambridge University Press

Table 3. Two news articles selected from TR-News and MLSum (TR)

	TR-News	MLSum (TR)
URL	https://www.haberturk.com/avrupa-birligi-abd-ve-cinli-teknoloji-devleriyle-mucadele-plani-hazirladi-2515715-teknoloji	https://www.internethaber.com/ise-surekli-gec-kalan-kadin-sorunun-kaynagini-bulunca-sok-oldu-2040194h.htm
Title	Avrupa Birliği ABD ve Çinli teknoloji devleriyle mücadele plan hazırladı	İşe sürekli geç kalan kadın, sorunun kaynağını bulunca şok oldu!
Abstract (Summary)	Avrupa Birliği Google, Microsoft, Apple gibi ABD'li ve Baidu, Alibaba gibi Çinli dev teknoloji şirketleriyle mücadele için bir plan hazırladı. Plan kapsamında kurulacak 100 milyar dolarlık fon Avrupalı teknoloji şirketlerine yatırım yapacak. Planda ayrıca Nokia'nın yıldızının parladığ yıllardaki stratejilerin uygulanması gerektiği belirtildi	Brezilya'da işe geç kalmaya başlayan bir kadın, alarmının her sabah kedisi tarafından kapatıldığını keşfetti.
Content (Text)	Avrupa Birliği (AB) yetkililerinin, ABD Başkanı Donald Trump'ın ticaret savaşları politikası ve ABD merkezli teknoloji devleri Google, Apple, Amazon, Microsoft ve Facebook'a karşı alınacak önlemler hakkında 173 sayfalık bir plan hazırladığı bildirildi. Politico'nun haberine göre, plan öncelikle bir Avrupa Gelecek Fonu kurulmasını öngörüyor. Söz konusu fonun gelecek vadeden Avrupalı firmalara 100 milyar dolar yatırım yaparak ABD'li ve Çinli teknoloji şirketlerine karşı denge oluşturması hedefleniyor. (. . .)	Brezilya'da Sao Paolo'da yaşayan bir kadın, işe sürekli geç kalmaya başlayınca bu durumun nedenini araştırmaya başladı. Sabah saatlerine kurduğu alarmı duymamaktan şikayetçi olan kadın, yaptığı araştırma sonucunda işe geç kalma sebebinin kedisi olduğunu keşfetti. Kadın tarafından kaydedilen görüntülerde, telefon alarmı çalmaya başladıktan sonra kedisinin telefonun yanına gelerek alarmı patisiyle kapattığı görülüyor. (. . .)
Topic	Teknoloji	–
Tags	['microsoft', 'apple', 'google', 'baidu', 'haberler']	–
Date	23.08.2019 - 13:01	00/06/2019
Author	DHA	–
Source	haberturk	–

total number of words. Agglutinative languages tend to have larger vocabulary sizes when compared to other languages due to the high number of suffixes the words can take. This can also be seen when the TTR values are compared; TR-News and MLSum (TR) have similar ratios, whereas CNN/Daily Mail and XSum have much lower ratios. Lastly, Combined-TR has a slightly lower TTR compared to TR-News and MLSum (TR) since its vocabulary size is less than the sum of those of the two datasets. Importantly, higher vocabulary size of Turkish brings more complexity and causes NLP tasks to become more challenging when compared to English (Oflazer 2014).

An example article from each dataset is given in Table 3. The table displays the fields URL, title, abstract, content, and date which are common to both datasets. However, TR-News also contains other valuable fields, which are topic, tags, author, and source but they are not relevant for this study and will not be used. The content field in the table has been cropped for convenience.

5. Experiments

In this section, we provide an analysis of the tokenization methods used in the models, briefly explain the main experiments, and present the results of the novelty analysis for the datasets used in this study. We focus on two different abstractive summarization tasks: text summarization and title generation. For both tasks, we make use of the state-of-the-art pretrained models and finetune them on the Turkish datasets.

https://doi.org/10.1017/S1351324922000195 Published online by Cambridge University Press

Table 4. Tokenization outputs of the methods for a given Turkish sentence which translates to "If one day, my words are against science, choose science"

Method	Output
	input: eğer bir gün benim sözlerim bilimle ters düşerse bilimi seçin
mT5	['_', 'eğer', '_ bir', '_ gün', '_ benim', '_ söz', 'lerim', '_ bilim', 'le', '_ ter', 's', '_ düş', 'erse', '_ bilim', 'i', '_ seçi', 'n', '.']
mBART	['_ eğer', '_ bir', '_ gün', '_ benim', '_ sözleri', 'm', '_ bilim', 'le', '_ ter', 's', '_ düş', 'er', 'se', '_ bilim', 'i', '_ seç', 'in', '.']
mBERT-uncased	['[UNK]', 'bir', '[UNK]', 'beni', '##m', '[UNK]', 'bilim', '##le', 'ter', '##s', '[UNK]', 'bilim', '##i', '[UNK]', '.']
mBERT-cased	['e', '##ğer', 'bir', 'gün', 'beni', '##m', 'söz', '##leri', '##m', 'bilim', '##le', 'ter', '##s', 'd', '##üş', '##erse', 'bilim', '##i', 'se', '##çi', '##n', '.']
BERTurk-uncased	['eğer', 'bir', 'gün', 'benim', 'sözleri', '##m', 'bilim', '##le', 'ters', 'düşer', '##se', 'bilimi', 'seçin', '.']
BERTurk-cased	['eğer', 'bir', 'gün', 'benim', 'sözleri', '##m', 'bilim', '##le', 'ters', 'düşer', '##se', 'bilimi', 'seçin', '.']

5.1 Tokenization analysis

Tokenization is one of the most important preprocessing steps in NLP problems. Tokenization approaches may vary depending on the problem. Simple methods such as whitespace tokenization can be applied if the vocabulary size is low, but in most cases the vocabulary size is immense. To solve the out-of-vocabulary problem, subword tokenization methods such as WordPiece (Wu *et al*. 2016), BPE (Sennrich *et al*. 2016), and SentencePiece (Kudo 2018) that can represent all the tokens with a vocabulary of a reasonable size have become popular in most sequence-to-sequence problems like machine translation and text summarization (Vaswani *et al*. 2017; Liu and Lapata 2019). The vocabulary and its size are critical because the input space and the output space of the pretrained models are directly determined by the tokenization method. This becomes even more important when the input is in a morphologically rich language such as Turkish or Czech and accordingly the input space has a much higher vocabulary size due to its nature.

In this work, we used BERT2BERT, mBART, and mT5 models where each model has been pretrained with a different tokenization method and has a different vocabulary. For BERT2BERT architecture, two BERT-based models were used: the multilingual BERT (mBERT) and the monolingual BERTurk. Similar to the majority of published research in the summarization literature, the inputs to all the models are given in lowercase. While converting into lowercase, we took into consideration a special case in Turkish: the lowercases of characters "İ" and "I" are, respectively, "i" and "ı", unlike the "I"-"i" combination in English. Since the inputs to the models are in lowercase, we have decided to use the lowercase variations of the BERT-based models, but have recognized some encoding problems with the mBERT-uncased model. Interestingly, the encoding problem was not present in the mBERT-cased model. Therefore, we decided to additionally use the cased versions of both mBERT and BERTurk (although uncased version of BERTurk does not have any encoding problems) to further evaluate its impacts.

To show the notable differences between the tokenizers, an example input sentence and the tokenized outputs under each tokenization method are displayed in Table 4. As can be seen, all models' tokenizers behave uniquely and have their own format when splitting the words into subwords. The models mT5 and mBART use the SentencePiece method and place an underscore between the words and do not place any special characters between the subwords. BERT-based methods, on the other hand, make use of the WordPiece tokenization method and only place "##" between the subwords. The tokenizers are specific to the models, and the outputs can differ based on the vocabulary size or the cased and uncased variation of the models. The outputs of the BERTurk models are more concise in terms of subwords, whereas the outputs of the multilingual

https://doi.org/10.1017/S1351324922000195 Published online by Cambridge University Press

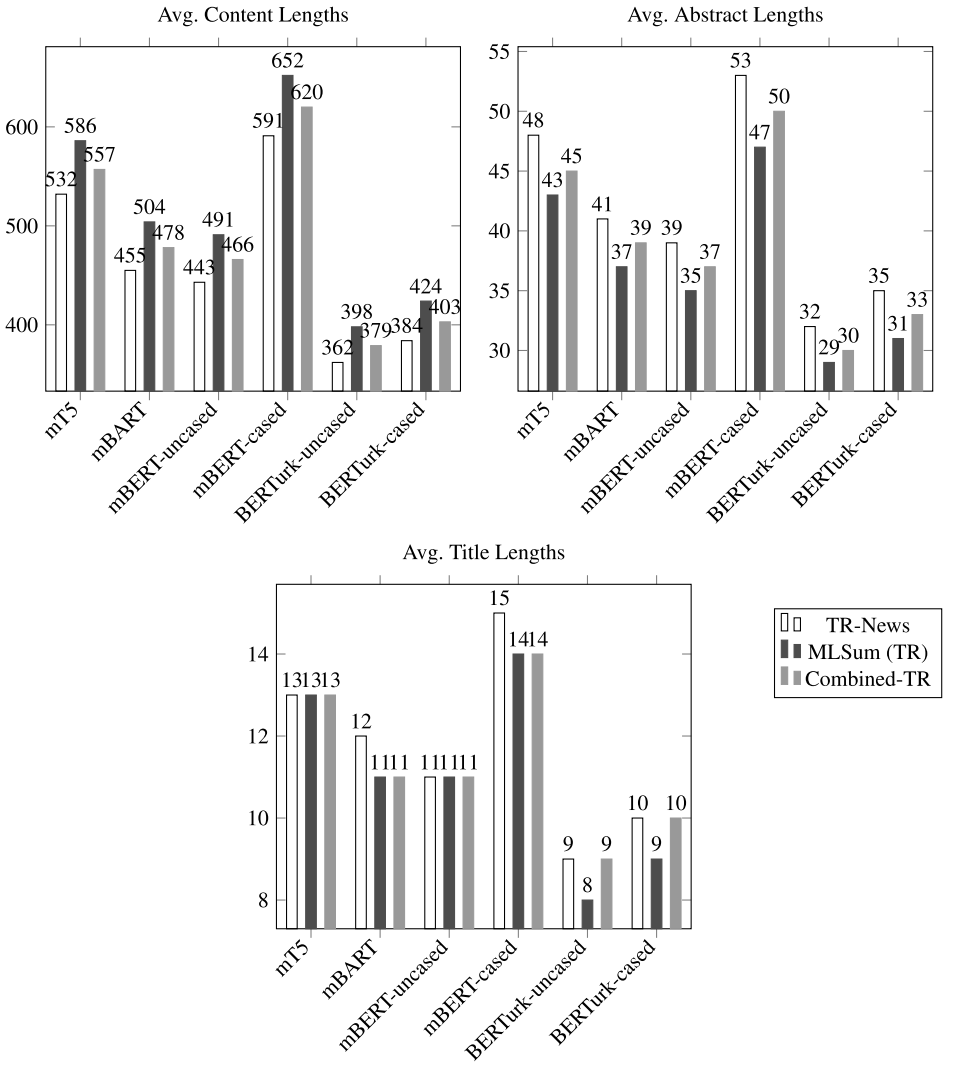

Figure 4. Average number of tokens generated by the tokenizers of the models for content, abstract, and title.

BERT model tend to be longer and have been split from grammatically unrelated parts of the words. The tokens' output for the mBERT-uncased model show the encoding problem discussed earlier where the words with some Turkish specific characters (e.g., ğ, ü, ö, ş, ç) cannot be covered within the model's tokenizer properly. As a result, each model's output varies in terms of subwords and the number of subwords. This might reflect to the downstream tasks in terms of performance (Rust *et al*. 2020).

The average number of tokens generated by the models' tokenizers is given in Figure 4. We see that multilingual models generate more tokens compared to monolingual BERTurk models for all three fields as exemplified in Table 4. The mBERT-cased model is the one that generates the most number of tokens, whereas the BERTurk-uncased model generates the least number of tokens. We believe the large difference between the cased and uncased versions of mBERT to be caused by the unknown tokens generated by the uncased variant. Additionally, a comparison between Table 1 and Figure 4 shows the gap between the number of words and subwords. All this information is important when constructing the models since the encoder and decoder lengths of the models

https://doi.org/10.1017/S1351324922000195 Published online by Cambridge University Press

need to be set based on these values. In this study, we consider the average and the maximum number of tokens generated by the tokenizers to determine an optimal size for the encoder and decoder lengths when finetuning the tasks.

5.2 Experiment 1—summary generation

The first experiment aims to produce news article summaries in an abstractive manner using the pretrained encoder-decoder networks. For this purpose, we employ the mT5, mBART, and BERT2BERT models. For BERT-based models, both multilingual BERT models and monolingual BERTurk models are utilized to measure the effectiveness of monolingual pretrained models on the news article summarization task. As stated earlier, the uncased variant of the multilingual model cannot tokenize properly the Turkish specific characters. To assess the impact of this problem, we used both variants in the experiments. Similarly, both cased and uncased variants of BERTurk are utilized. In all the BERT2BERT models, we make use of the same BERT model in the encoder and the decoder parts. Lastly, it is known that using more data when training deep learning models usually tends to result in better performances (Ng *et al*. 2014). We further investigate this notion with the same set of models on the Combined-TR dataset which we have created by merging the TR-News and MLSum (TR) datasets.

For the mT5 and mBART models, we set the maximum encoder length to 768 and the maximum decoder length to 128 based on the observations given in Figure 4 to cover most of the contents and abstracts of the documents. For the BERT-based models, the maximum encoder length is limited to 512 due to model restrictions and the maximum decoder length is set to 128 as in the other models. The Adafactor optimizer (Shazeer and Stern 2018) is used for the mT5 as suggested by the authors. In our early experimen,ts we also tried using the Adam optimizer (Kingma and Ba 2015) but we noticed that Adafactor converges much faster. The BERT-based models and the mBART model use the Adam optimizer. The learning rates for the mT5 and the other models are 1e-3 and 5e-5, respectively. During inference, we make use of tri-gram blocking to reduce the number of repetitions in the generated text.

Tesla V100 GPUs were used in the finetuning process of all the models with an effective batch size of 32. The models were finetuned for a maximum of 10 epochs; however, early-stopping with patience 2 was employed based on the validation loss. The number of warmup steps was set to 1000. Huggingface's transformers library was used for finetuning the models (Wolf *et al*. 2020).

5.3 Experiment 2—title generation

In the second experiment, we aim to generate news article titles in an abstractive manner using the same set of models and datasets as in the first experiment. Title generation task is also a summarization task in the sense that the model receives an input text that briefly describes the news article and a title that is suitable to the news is expected as an output. In this work, two types of inputs are used to generate the titles:

- **Abstract as input**: The reference summaries are considered to be concise representations of the news articles and are present for all the datasets used in this study. Therefore, we frame the title generation task by considering the abstract/reference summary as the input and the title as the output to the encoder-decoder model.
- **LEAD-3 as input**: In the literature, selecting the first three sentences of a news article (LEAD-3) is considered to be a strong baseline for the news article summarization task and is accordingly seen as a reference summary capable of reflecting the content of the news article (Narayan *et al*. 2018a). Hence, we use the LEAD-3 as a possible input to the title generation task as well.

https://doi.org/10.1017/S1351324922000195 Published online by Cambridge University Press

Table 5. Novelty ratios of the datasets with respect to the summary generation and title generation tasks. N1, N2, and N3 denote uni-gram, bi-gram and tri-gram ratios, respectively

	TR-News			MLSum (TR)			Combined-TR		
Tasks	N1	N2	N3	N1	N2	N3	N1	N2	N3
Summary	31.50	57.26	66.02	31.27	55.09	63.77	31.40	56.28	65.00
Title (Abstract)	52.61	65.11	59.56	48.22	66.21	67.88	50.62	65.61	63.33
Title (LEAD-3)	57.05	70.12	62.43	55.97	71.81	71.01	56.56	70.89	66.33

For this experiment, the maximum encoder and decoder lengths have been set to 256 and 64, respectively. The remaining parameters and settings for all the models are the same as the first experiment.

5.4 Novelty analysis

In abstractive summarization, it is important to assess the degree of abstractiveness (text novelty) of the reference summaries in the datasets and of the generated summaries. High level of abstractiveness of the summaries in a dataset can be interpreted as being more challenging for the summarization task. In addition to being able to generate concise, relevant, and fluent summaries as in extractive models, abstractive models are also responsible for generating summaries that are genuine which do not contain a high amount of copied words from the source article. Novelty ratio is a commonly used metric which can provide insight to how abstractive a summary of a given article is. The novelty ratio is calculated as the percentage of the number of words in the summary that do not occur in the source document. To observe the abstractiveness of the datasets used in this study, we calculated the novelty ratios of the reference summaries and the titles. Table 5 shows the novelty ratios in terms of n-grams. For title generation, novelty ratios were calculated separately for the abstract and the LEAD-3 sentences as the source document. As can be seen, TR-News is slightly more abstractive than MLSum (TR) in terms of the summary generation task. For the title generation tasks, TR-News seems to have higher uni-gram ratios but lower bi-gram and tri-gram ratios compared to MLSum (TR). The novelty analysis of the generated summaries and titles will be given in Section 6.1.3.

6. Results

In this section, we evaluate our findings both quantitatively and qualitatively for both the summary generation and the title generation tasks.

6.1 Quantitative results

The models described in Section 3 were evaluated using the experimental settings discussed in the previous section with the ROUGE metric (Lin 2004), a commonly used evaluation metric in text summarization. ROUGE-1, ROUGE-2, and ROUGE-L scores are reported. The ROUGE-n score measures the informativeness of the generated summaries by counting the number of common n-grams between the generated summary and the reference summary. ROUGE-L calculates the number of overlapping n-grams based on the longest common sub-sequences and measures the fluency of the generated summaries. In addition to the ROUGE metrics, the novelty ratios of the generated summaries and titles are also calculated in terms n-grams ($n = 1, 2, 3$).

https://doi.org/10.1017/S1351324922000195 Published online by Cambridge University Press

Table 6. Text summarization results of pretrained encoder-decoder models on TR-News, MLSum (TR), and Combined-TR datasets. ROUGE-1 (R1), ROUGE-2 (R2), and ROUGE-L (RL) scores are given in F-measure. "-" denotes result is not available. Bold values show the highest scores obtained in the experiments per dataset

	TR-News			MLSum (TR)			Combined-TR		
Models	R1	R2	RL	R1	R2	RL	R1	R2	RL
LEAD-2	31.37	17.91	26.92	36.32	23.18	31.39	33.61	20.31	28.95
LEAD-3	28.64	16.21	24.07	34.88	22.20	29.45	31.47	18.93	26.51
Pointer generator (See *et al*. 2017)	31.61	18.55	29.57	38.04	25.01	35.70	35.23	22.03	33.04
(Scialom *et al*. 2020)	–	–	–	36.90	21.77	32.60	–	–	–
mT5	**41.13**	**25.75**	37.60	**42.26**	**27.81**	**37.96**	42.49	27.58	38.67
mBART	40.52	25.22	36.80	40.47	26.17	36.22	41.97	26.95	38.08
BERTurk-uncased	40.50	25.24	37.23	41.47	27.31	37.52	42.51	27.62	38.86
BERTurk-cased	41.06	25.60	**37.69**	41.48	27.23	37.66	**42.75**	**27.83**	**39.08**
mBERT-uncased	33.04	14.94	30.42	33.59	15.98	30.51	34.13	15.95	31.20
mBERT-cased	39.73	24.51	36.37	40.27	26.22	36.40	41.20	26.35	37.50

6.1.1 Experiment 1—summary generation

Table 6 shows the results for the first experiment. In the first part of the table, the performance of the LEAD-2 and LEAD-3 baselines is given for all the datasets. LEAD baselines are commonly referred to in the evaluation of text summarization studies and are considered to be hard baselines to surpass (Torres-Moreno 2014; Narayan, Cohen, and Lapata 2018b). We also provide the results for the pointer-generator network (See *et al*. 2017) which are the state-of-the art results for both datasets. The second part of the table displays the results for the pretrained encoder-decoder models used in this study.

It is apparent that the mT5 and BERTurk-cased models perform very close to each other where the mT5 model being better on the individual datasets. Importantly, all the models except mBERT-uncased outperformed the pointer-generator results with a large margin on both the TR-News and MLSum (TR) datasets. Hence, the results show that the pretrained encoder-decoder networks perform better than the RNN-based method (pointer-generator network) for the Turkish language (RQ1). Another finding for all the pretrained encoder-decoder models is the improvement obtained by joining the two datasets (Combined-TR). Increasing the number of training samples for the summary generation task seems to substantially increase the efficacy of all the models (RQ3). This supports the common knowledge of obtaining more training data would usually lead to performance gains in deep neural network based models. Additionally, the BERTurk-cased model slightly outperforms mT5 on the Combined-TR dataset. The multilingual BART model has performed worse than the BERTurk and mT5 models, but better than the multilingual BERT models for all the datasets.

When the BERT2BERT models are compared within themselves, it is evident that the cased models tend to perform better than the uncased models for both BERTurk and multilingual BERT. For multilingual BERT, this is mostly due to the encoding problem. Moreover, the monolingual BERT models outperformed the multilingual BERT models, showing the effectiveness of pretraining on language specific data (RQ2).

6.1.2 Experiment 2—title generation

The second experiment aims to measure the performance of the models on the title generation task. We use two different input types: abstract and LEAD sentences. Table 7 shows the results

https://doi.org/10.1017/S1351324922000195 Published online by Cambridge University Press

Table 7. Title generation (abstract as input) results of pretrained encoder-decoder models on TR-News, MLSum (TR), and Combined-TR datasets. ROUGE-1 (R1), ROUGE-2 (R2), and ROUGE-L (RL) are given in F-measure. Bold values show the highest scores obtained in the experiments per dataset

	TR-News			MLSum (TR)			Combined-TR		
Models	R1	R2	RL	R1	R2	RL	R1	R2	RL
mT5	**41.87**	**24.49**	40.87	**40.77**	**22.42**	**38.97**	43.04	**25.14**	41.59
mBART	37.72	20.99	36.74	34.85	18.03	33.46	39.94	22.44	38.46
BERTurk-uncased	40.93	23.67	40.05	38.04	20.16	36.37	42.48	24.51	41.07
BERTurk-cased	41.87	24.37	**40.88**	39.35	21.14	37.55	**43.06**	25.13	**41.61**
mBERT-uncased	33.88	15.39	33.20	31.18	12.68	30.04	34.48	15.46	33.50
mBERT-cased	40.83	23.50	39.89	38.98	21.07	37.30	42.14	24.32	40.70

Table 8. Title generation (LEAD-3 as input) results of pretrained encoder-decoder models on TR-News, MLSum (TR), and Combined-TR datasets. ROUGE-1 (R1), ROUGE-2 (R2), and ROUGE-L (RL) are given in F-measure. Bold values show the highest scores obtained in the experiments per dataset

	TR-News			MLSum (TR)			Combined-TR		
Models	R1	R2	RL	R1	R2	RL	R1	R2	RL
mT5	**34.89**	**18.58**	34.01	**32.15**	**16.29**	**30.75**	35.53	**19.14**	34.36
mBART	31.81	15.96	31.03	27.06	13.06	25.92	23.27	11.00	22.44
BERTurk-uncased	33.80	17.58	33.06	30.31	15.05	29.11	34.72	18.42	33.66
BERTurk-cased	34.84	18.31	**34.08**	31.99	16.05	30.58	**35.66**	19.10	**34.52**
mBERT-uncased	27.26	11.29	26.72	24.73	9.49	23.88	28.05	11.64	27.27
mBERT-cased	33.28	17.17	32.44	30.79	15.47	29.52	34.35	18.19	33.26

where abstract is used as the input and Table 8 shows the results where LEAD-3 is used as the input. The results are in parallel to the summary generation task in terms of model performances. When abstract is given as input to the models, mT5 and BERTurk-cased perform very close to each other in all the datasets, where the mT5 model performs slightly better on the MLSum (TR) dataset. The same is true for the LEAD-3 case in Table 8. In addition, combining the datasets has shown performance gains for all the models (RQ3). In terms of the BERT2BERT models, cased models have again shown to be better than their uncased variations. Moreover, monolingual BERT models outperformed their multilingual variants on the title generation task regardless of the input types (RQ2).

Interestingly, the mBART model has been unstable during training and this is reflected in the results. For instance, the model has shown an unexpectedly low performance on the Combined-TR dataset as seen in Table 8. If the mBERT-uncased model is set aside due to encoding problems, mBART can be considered as the model with the poorest performance amongst all the models used in this study for the title generation task. The mBART model seems to have performed worse compared to the summary generation task. This might indicate that mBART might be more suitable for tasks that require longer inputs and outputs.

Another important finding for the title generation task is the impact of the input. When Tables 7 and 8 are compared, providing the abstract as input to the title generation task seems to be more effective (RQ5). There can be several reasons behind this difference: (1) abstract is a

https://doi.org/10.1017/S1351324922000195 Published online by Cambridge University Press

Table 9. Title generation LEAD sentences ablation study results. ROUGE-1 (R1), ROUGE-2 (R2), and ROUGE-L (RL) are given in F-measure

	TR-News			MLSum (TR)			Combined-TR		
Models	R1	R2	RL	R1	R2	RL	R1	R2	RL
LEAD-1	28.83	14.16	28.11	27.60	13.81	26.59	29.82	15.26	28.91
LEAD-2	33.20	17.31	32.39	30.78	15.64	29.51	33.84	17.87	32.75
LEAD-3	34.89	18.58	34.01	32.15	16.29	30.75	35.53	19.14	34.36
LEAD-4	35.41	18.95	34.54	33.06	17.06	31.56	36.06	19.48	34.86
LEAD-5	35.70	19.18	34.78	33.31	17.26	31.83	36.72	20.03	35.50

more informative summary compared to LEAD-3, (2) abstract contains keywords more similar to the title, (3) abstract (around 1.5 sentences for both datasets—see Table 1) being shorter than LEAD-3 (3 sentences) holds more relevant data for the title. To find out the impact of the input length related to the third claim, we conducted an ablation study where the first n LEAD sentences are given as input to the title generation model. For this and the other ablation studies in this paper, the mT5 model is selected since it has shown to be one of the best models for both summary generation and title generation tasks. A total of five models were trained by feeding the first n sentences from the content as input expecting the title to be generated in the output. Table 9 shows the results for the ablation study. It can be seen that increasing the number of sentences in the input seems to increase the performance in the title generation task for all the datasets. This ablation study concludes that the length of the input is not a relevant factor that can explain the performance difference between providing the abstract versus LEAD-3 as the input.

6.1.3 Novelty analysis

As explained in Section 5.4, the novelty metric is used to assess the generated text in terms of abstractiveness. In this section, we evaluate the novelty degree of the generated summaries and titles in terms of n-grams (uni-gram, bi-gram, and tri-gram) on all the datasets. Tables 10 and 11 show the novelty results for the summary generation and the title generation tasks, respectively. It is seen that the BERTurk models produce more novel outputs in both tasks, whereas the mT5 and the mBART models tend to produce less abstractive outputs compared to the other models

Table 10. Novelty ratios of the summaries generated by the models per dataset. N1, N2, and N3 denote uni-gram, bi-gram, and tri-gram ratios, respectively. Bold values show the highest scores obtained in the experiments per dataset (the mBERT-uncased results are misleading and are ignored due to the high number of unknown tokens output)

	TR-News			MLSum (TR)			Combined-TR		
Tasks	N1	N2	N3	N1	N2	N3	N1	N2	N3
mT5	9.87	21.23	27.48	9.90	20.38	26.41	10.51	21.44	27.21
mBART	9.98	21.08	26.72	8.06	17.63	23.08	11.31	23.30	29.18
BERTurk-uncased	15.24	32.89	42.01	**16.23**	**32.63**	**41.27**	14.17	29.76	37.88
BERTurk-cased	**16.44**	**35.08**	**44.43**	15.93	32.21	40.72	**15.26**	**31.48**	**39.67**
mBERT-uncased	12.24	45.68	60.28	12.53	45.12	59.67	12.57	45.16	59.63
mBERT-cased	14.17	31.19	40.35	15.18	30.75	39.01	13.89	29.36	37.48

https://doi.org/10.1017/S1351324922000195 Published online by Cambridge University Press

Table 11. Novelty ratios of the titles (abstracts are given as input) generated by the models per dataset. N1, N2, and N3 denote uni-gram, bi-gram, and tri-gram ratios, respectively. Bold values show the highest scores obtained in the experiments per dataset

	TR-News			MLSum (TR)			Combined-TR		
Tasks	N1	N2	N3	N1	N2	N3	N1	N2	N3
mT5	29.70	47.78	50.23	33.06	52.13	55.53	29.41	48.10	51.32
mBART	27.18	41.96	43.12	35.86	51.25	49.96	33.70	49.84	51.21
BERTurk-uncased	37.42	55.54	**55.45**	**47.00**	**64.29**	**64.55**	**35.63**	**54.71**	**56.38**
BERTurk-cased	**37.56**	**55.67**	55.43	44.81	62.16	62.60	34.43	53.32	55.49
mBERT-uncased	33.04	53.92	48.72	43.18	61.39	54.85	32.11	54.67	50.73
mBERT-cased	31.48	50.60	52.90	37.22	55.63	58.23	30.98	50.42	53.36

for all the datasets. It is important to note that the results for mBERT-uncased are misleading due to a high number of unknown ([UNK]) tokens generated in the outputs, which is caused by the character encoding problem of the model. Especially, the bi-gram and the tri-gram results of mBERT-uncased for the summarization task point out to irregular increases compared to its cased version. Hence, we choose to ignore the mBERT-uncased results for the novelty analysis. Lastly, the novelty ratios for the title generation task are much higher compared to summary generation. This shows that as the length of the outputs gets longer, the novelty ratio decreases.

6.1.4 Cross-dataset evaluations (RQ4)

In Experiment 1 and Experiment 2, the models have been trained and evaluated on the same dataset. However, in real-world applications of such models, one cannot make the assumption that the data will always come from the same distribution or source. Models in general tend to perform worse on sources which they were not trained on. In this experiment, we aimed to observe whether evaluating the trained models across different datasets would lead to a significant amount of performance degradation. Since the datasets we use have statistically similar attributes as described in Section 4, we conducted cross-dataset evaluations on the two datasets and the combined dataset for both summary and title generation tasks. We made use of the mT5 model for all the evaluations.

Table 12 shows the results of cross-dataset evaluations. (For more detailed results, please see Tables A1 and A2 in the Appendix.) The rows correspond to the training set and the columns correspond to the test set. For the summary generation task, the best results were obtained with the mT5-Combined-TR model (the mT5 model trained on the Combined-TR dataset). The performance of this model outperforms all the results obtained with the other two models (training

Table 12. Cross-dataset evaluation results for the summary generation and the title generation (abstract as input) tasks. The values correspond to ROUGE-1 scores

	Summary			Title (Abstract)		
Model & training set	TR-News	MLSum (TR)	Combined-TR	TR-News	MLSum (TR)	Combined-TR
mT5-TR-News	41.13	40.99	41.06	41.87	41.81	41.84
mT5-MLSum-TR	37.25	42.26	39.52	36.32	40.77	38.34
mT5-Combined-TR	41.23	44.01	42.49	42.46	43.79	43.04

https://doi.org/10.1017/S1351324922000195 Published online by Cambridge University Press

sets) regardless of the test set used. This observation supports the findings of the previous two experiments related to RQ3.

When we consider training on individual datasets, we see that the models trained on TR-News and MLSum (TR) perform the best on their own test sets. However, the performance of mT5-TR-News is slightly affected when the test set changes, whereas the performance of mT5-MLSum-TR drops up to 5 ROUGE-1 points. The mT5-TR-News model also gives higher score than the mT5-MLSum-TR model on the combined test set. Lastly, the mT5-Combined-TR model performs better on the test set of MLSum (TR) rather than the combined test set. All these observations imply that the model trained on TR-News is a more robust model and performs well on data from other sources. This might indicate that TR-News is a more diverse dataset, providing richer information. On the other hand, the models trained on MLSum-TR and the combined training sets perform much better when the data come from the MLSum-TR source. This is probably a signal about the more specific nature of the MLSum (TR) dataset.

The results for the title generation task also support the cross-dataset findings of the summary generation task. In a similar manner, the mT5-Combined-TR model achieves the highest ROUGE-1 score across all the datasets. The model trained on MLSum (TR) struggles on the TR-News and Combined-TR datasets compared to the other models and also obtains the lowest score on its own test set.

6.1.5 Generation parameters (Beam size and early-stopping)

In the encoder-decoder models, during the inference phase the outputs are generated in an auto-regressive manner. Each token that is output is fed to the decoder as input in the next decoding step. Hence, each output token affects the tokens that will be generated in the future, which makes the decoding strategy an important variable that determines the quality of the generated text. In text summarization, the most commonly used decoding strategy at inference time is beam search. The aim of beam search is to keep track of the best n hypotheses at each step so that the sequence with the highest overall probability is not eliminated at an early stage due to a low probability token. The number n plays an important role in the performance of beam search. In this respect, we aimed to assess the effect of beam search with various beam sizes (1–4) where beam size 1 refers to greedy search. Moreover, we investigate the use of the early-stopping mechanism during the decoding phase, which allows the decoder to stop when all the hypotheses reach the special end of sentence token ([EOS]) instead of continuing until the predefined decoding length.

Table 13 shows the ROUGE-1 scores for the summaries and titles generated using the mT5 model on all the datasets. (For more detailed results, please see Tables A3 and A4 in the Appendix.) We see that increasing the beam size mostly increases the performance. Although increasing the

Table 13. Results for the summary generation and title generation (abstract as input) tasks with various beam sizes and early-stopping method. The values correspond to ROUGE-1 scores. Bold values show the highest scores obtained in the experiments per dataset

	Summary			Title (Abstract)		
Parameters	TR-News	MLSum (TR)	Combined-TR	TR-News	MLSum (TR)	Combined-TR
mT5-beam-1	40.74	40.87	41.99	40.41	37.93	41.12
mT5-beam-2	**41.34**	42.13	**42.61**	41.58	39.95	42.49
mT5-beam-3	41.30	42.18	42.59	41.82	40.54	42.91
mT5-beam-4	41.13	**42.26**	42.49	**41.87**	**40.77**	**43.04**
mT5-beam-4 & early-stopping	41.15	41.36	42.18	41.66	40.04	42.53

https://doi.org/10.1017/S1351324922000195 Published online by Cambridge University Press

Table 14. ROUGE scores calculated with different preprocessing settings. "Punct removed" refers to removing the punctuations, whereas "Punct kept" refers to keeping the punctuations before the ROUGE calculations. "Stems taken" refers to applying stemming operation on the words, whereas "Stems not taken" refers to leaving the words in their surface form before the ROUGE calculations

	Summary			Title (Abstract)		
Parameters	TR-News	MLSum (TR)	Combined-TR	TR-News	MLSum (TR)	Combined-TR
Punct removed & stems taken	41.13	42.26	42.49	41.87	40.77	43.04
Punct removed & stems not taken	37.60	39.03	39.12	37.91	37.22	39.24
Punct kept & stems taken	43.64	44.60	44.83	40.00	39.23	41.09
Punct kept & stems not taken	40.55	41.76	41.88	36.35	35.92	37.56

beam size past the size of 4 might continue increasing the scores, such an option brings more complexity and computational time. Also, we see that in some cases the ROUGE gains start to decrease after the beam size of 3. Based on these results, we consider the beam size of 4 as both yielding high ROUGE scores and allowing computationally tractable inference. Lastly, early-stopping is employed on the configuration with beam size of 4, but it reduced the performance in nearly all the evaluations.

6.1.6 ROUGE assessment variations

ROUGE (Lin 2004) is the most commonly used set of evaluation metrics in the literature for text summarization. The calculations are based on the overlapping tokens between the reference and the system summaries. Hence, the metric in its essence is based on exact match of the tokens. Therefore, any change to the tokens in the reference and the system summaries in terms of preprocessing operations before evaluating the ROUGE scores will affect the results. Removing the punctuations and applying stemming are commonly used as preprocessing operations in ROUGE evaluations. However, in most publications these details are not shared which makes interpreting the results difficult in some cases. Although stemming does not have a high impact on the results in English, it alters the surface form of an important number of words in agglutinative languages like Turkish, causing a significant change in the ROUGE scores.

Consequently, we aimed to show that such preferences can impact the results. We held a set of experiments which show the effect of punctuations and the stemming operation in ROUGE evaluations. During the experiments, we realized that the original ROUGE script which is implemented in Perl and known as ROUGE 1.5.5[g] is not capable of correctly processing non-English characters. Therefore, we made use of another repository which replicates the original Perl script in the Python programming language.[h] However, several modifications were needed in order to make it compatible with Turkish so that we made the necessary changes and also integrated Turkish stemming.[i]

Table 14 depicts the ROUGE-1 results obtained for both summary and title generation tasks. (For more detailed results, please see Tables A5 and A6 in the Appendix.) As can be seen from the results, applying stemming highly increases the ROUGE-1 scores for both tasks on all the datasets. This is expected for the Turkish language since the amount of agglutination is very high. Keeping the punctuations seems to increase the score for the summary generation task as opposed to the title generation task. This implies that as the length of the evaluated texts gets longer, the amount of punctuations that get overlapped also increases, thus improving the ROUGE-1 score.

[g] https://github.com/andersjo/pyrouge/tree/master/tools/ROUGE-1.5.5.
[h] https://github.com/google-research/google-research/tree/master/rouge.
[i] https://github.com/otuncelli/turkish-stemmer-python.

https://doi.org/10.1017/S1351324922000195 Published online by Cambridge University Press

Furthermore, we held an additional set of experiments to observe whether the performance rankings of the models get affected by the preprocessing operation. Accordingly, the same experiment was conducted on all the models and datasets used in this study for the text summarization task. The results (Tables A7, A8, and A9 in the Appendix) are in parallel with the findings in Table 6 where mT5 and BERTurk models were again superior to the other models in most settings. However, for TR-News and Combined-TR we observe that the performance rankings in some cases change depending on the choice of the parameters. For instance, the setting where stemming is not applied and punctuations are not kept in Table A7, BERTurk-cased slightly passes the mT5 model on the TR-News dataset. For the experiment in Table A9, the best model becomes mT5 under the settings where punctuations are kept. These findings also support the claim that such preprocessing operations in ROUGE calculations can impact the results.

6.2 Qualitative results

Apart from quantitative analysis, we provide a qualitative analysis for both the text summarization and the title generation tasks. Although quantitative analysis gives an idea about the informativeness and fluency of the models, other important aspects such as coherence and cohesion are left out. In this respect, we examined randomly chosen 50 examples from each dataset (100 examples in total) to observe on real data how well the generated summaries and titles fit to the reference summaries and titles. In this section, we provide two illustrative examples for each task from each dataset that are interesting and challenging.

6.2.1 Summary generation

Tables 15 and 16 show an example from, respectively, TR-News and MLSum (TR) for the text summarization task. The words in bold denote novel unigrams (unigrams which are not present in the input text) generated by the models, whereas the underlined texts are for reference in the discussion below. For both examples, the content, the reference summary, and the generated summaries of the models are given. All the texts in the tables except the content fields have been translated to English. The content fields were left out due to limited space.

The first example in Table 15 is correctly summarized by all the models, except the mBERT-uncased model. Almost all summaries are very extractive; some of the summaries directly copy the first sentence and most of the models are not able to produce any novel unigrams. The most abstractive summaries belong to mT5 and BERTurk-cased, and they are very similar to each other. BERTurk-uncased changed the sentence from active voice to passive voice rather than directly copying, which made the summary more abstractive. The mBERT-cased model left the word "da" (also) when copying from the first sentence, but this slight change corrupted the meaning of the sentence. On the other hand, the mBERT-uncased model failed to output words with Turkish specific characters ("fenerbahçe'nin," "ettiği," etc.) which caused the summary to be incorrect both syntactically and semantically.

Summaries generated for the second example are given in Table 16. None of the models included the number of towns and villages as in the reference summary or their names in the generated summaries. The mT5 model produced a correct but incomplete summary by not specifying the damaged decares of land. The mBART model generated a token referring to an unspecified location "istan" which does not exist in the news article. The BERTurk-uncased model has output unsupported information by emphasizing that many homes and businesses were affected by the flood. All the models except mBART and mBERT-uncased managed to produce the word "Niğde'de" (in Niğde) which is an important novel word present in the reference summary. The best summary in terms of completeness is produced by the BERTurk-cased model although containing a small grammatical error "arazileri" (lands) (it should have been "arazilerini" by taking

https://doi.org/10.1017/S1351324922000195 Published online by Cambridge University Press

Table 15. An example from the test set of TR-News accompanied with the summaries generated by the models. The spelling and grammatical errors in the original texts are left as is. News article's content is given as the input, and the reference summary is the abstract of the article. The words in bold denote novel unigrams (unigrams which are not present in the input text) generated by the models, whereas the underlined texts are for reference in the discussion

Content	fenerbahçe'nin braga'ya 4-1 yenilerek avrupa'ya veda ettiği maçı yöneten hakem ivan bebek'i hırvat basını da ağır dille eleştirdi. hırvat basınında yer alan bir analizde, ivan bebek'in sorgulanabilir kararlarla 3 fenerbahçeli oyuncuyu kırmızı kartla oyundan ihraç ettiğini, volkan şen'in ironik bir şekilde hakemi alkışladıktan sonra ittiğini belirtti. 67. dakikadaki penaltı kararının çok ağır ve haksız olduğu belirtilirken, bu dakikadan sonra ivan bebek'in fenerbahçe için kabus gibi bir yönettiğini, bu dakikadan sonra sarı lacivertlilerin kontrolü kaybettiğinin altı çizildi. ayrıca, yapılan analizde ivan bebek'in bu maçta verdiği kararların daha sonra da çok tartışılacağı kaydedildi. (sporx) (EN: the croatian press also heavily criticized referee ivan bebek, who directed the match in which fenerbahçe lost 4-1 to braga and bid farewell to europe. in an analysis in the croatian press, it was stated that ivan bebek dismissed 3 fenerbahçe players with a red card with questionable decisions, and volkan şen ironically pushed the referee after applauding. while stating that the penalty decision in the 67th minute was very harsh and unfair, it was underlined that after this minute, ivan bebek was leading like a nightmare fenerbahçe, and after this minute, the yellow blue colored team lost control. additionally, it was noted in the analysis that the decisions made by ivan bebek in this match will be discussed a lot later on. (sporx))
Reference (Abstract)	hırvat hakem ivan bebek, fenerbahçe'nin braga'ya 4-1 yenilerek elendiği maçta verdiği ve es geçtiği kararlarla kendi ülkesinde de ağır eleştilere maruz kaldı. (EN: croatian referee ivan bebek was also subjected to heavy criticism in his own country for the decisions he made in the match where fenerbahçe was defeated 4-1 and eliminated by braga.)
mT5	**braga-fenerbahçe maçını** yöneten hakem ivan bebek'i hırvat basını da ağır dille eleştirdi. (EN: croatian press also heavily criticized the referee, ivan bebek, who directed the braga-fenerbahçe match.)
mBART	fenerbahçe'nin braga'ya 4-1 yenilerek avrupa'ya veda ettiği maçı yöneten hakem ivan bebek'i hırvat basını da ağır dille eleştirdi. (EN: croatian press also heavily criticized the referee ivan bebek, who directed the match in which fenerbahçe lost 4-1 to braga and bid farewell to europe.)
BERTurk-uncased	fenerbahçe'nin braga'ya 4 - 1 yenilerek avrupa'ya veda ettiği maçı yöneten hakem ivan bebek, hırvat basınında da ağır dille **eleştirildi**. (EN: referee ivan bebek, who directed the match where fenerbahçe lost 4 - 1 to braga and bid farewell to europe, was also heavily criticized in the croatian press.)
BERTurk-cased	**braga-fenerbahçe maçını** yöneten ivan bebek'i hırvat basını da ağrı dille eleştirdi. (EN: croatian press also heavily criticized ivan bebek, who directed the braga - fenerbahce match.)
mBERT-uncased	braga'ya 4 - 1 yenilerek avrupa'ya veda hakem ivan bebek'i hırvat basını. (EN: beat braga 4 - 1, farewell to europe referee ivan bebek croatian press.)
mBERT-cased	fenerbahçe'nin braga'ya 4 - 1 yenilerek avrupa'ya veda ettiği maçı yöneten hakem ivan bebek'i hırvat basını ağır dille eleştirdi. (EN: croatian press heavily criticized the referee ivan bebek, who directed the match in which fenerbahçe lost 4 - 1 to braga and bid farewell to europe.)

the accusative form of the word). Lastly, the mBERT-uncased model once again failed to output a grammatically and semantically correct summary.

6.2.2 Title generation

Tables 17 and 18 show an example from, respectively, TR-News and MLSum (TR) for the title generation task. The words in bold denote novel words generated by the models, whereas the underlined texts are for reference in the discussion below. For both examples, the abstract, the reference title, and the generated titles of the models are given with their English translations.

https://doi.org/10.1017/S1351324922000195 Published online by Cambridge University Press

Table 16. An example from the test set of MLSum (TR) accompanied with the summaries generated by the models. News article's content is given as the input and the reference summary is the abstract of the article. The words in bold denote novel unigrams (unigrams which are not present in the input text) generated by the models, whereas the underlined texts are for reference in the discussion

Content	dün akşam saatlerinde etkili olan dolu ve sağanak nedeniyle merkeze bağlı konaklı, edikli beldeleri ile karaatlı ve çavdarlı köylerinde patates, fasulye ve mısır tarlalarını su bastı. tarım ve orman il müdürlüğü ekipleri bölgede yaptıkları incelemede yaklaşık 20 bin dekar alanın zarar gördüğünü tespit etti. tarım ve orman il müdürü asım baş, "niğde merkeze bağlı konaklı, edikli beldeleri ile karaatlı ve çavdarlı köylerinde yağmurun ardından dolu yağması bölgenin önemli geçim kaynaklarından başta patates, fasulye ve mısır olmak üzere bazı tarım ürünlerinin ekili olduğu arazileri olumsuz etkiledi. müdürlüğümüze bağlı ekiplerimizin şu anda tarım arazilerinde zarar tespit çalışmaları devam etmektedir" dedi (EN: due to the hail and heavy rain that was effective yesterday evening, potato, bean and corn fields in the towns of konaklı, edikli and karaatlı and çavdarlı villages were flooded. the teams of the provincial directorate of agriculture and forestry determined that approximately 20 thousand decares of land had been damaged in their examination in the region. asım baş, the provincial director of agriculture and forestry, said: "the hail after the rain in the konaklı, edikli towns and karaatlı and çavdarlı villages of the center of niğde adversely affected the lands where some agricultural products are cultivated, especially potatoes, beans and corn, which are important sources of income in the region. at the moment, damage assessments of our teams are ongoing in agricultural lands.")
Reference (Abstract)	niğde'de dün etkili olan sağanak ve dolu nedeniyle 4 belde ve köyde yaklaşık 20 bin dekar tarım arazisi zarar gördü. (EN: due to the heavy rain and hail that was effective yesterday in niğde, approximately 20 thousand decares of agricultural land in 4 towns and villages were damaged.)
mT5	**niğde'de** dün akşam saatlerinde etkili olan dolu ve sağanak nedeniyle patates, fasulye ve mısır tarlalarını su bastı. (EN: potato, bean, and corn fields were flooded in niğde due to the hail and heavy rain that were effective yesterday evening.)
mBART	**istan ilçesinde** etkili olan dolu ve sağanak nedeniyle 20 bin dekar alanın zarar gördüğü **bildirildi.** (EN: it has been reported that 20 thousand decares of land have been damaged due to the hail and heavy rain that were effective in the district of istan.)
BERTurk-uncased	**niğde'de** etkili olan dolu ve sağanak nedeniyle **çok sayıda ev** ve **iş yerini** su bastı. (EN: many homes and businesses were flooded due to the hail and heavy rain in niğde.)
BERTurk-cased	**niğde'de** dün akşam saatlerinde etkili olan dolu ve sağanak nedeniyle ekili tarım arazileri su bastı, 20 bin dekar alan zarar **gördü**. (EN: cultivated agricultural lands were flooded and 20 thousand decares of land were damaged in niğde due to the hail and heavy rain that were effective yesterday evening.)
mBERT-uncased	' da etkili olan dolu ve nedeniyle merkeze konaklı, edikli beldeleri ile karaatlı ve patates, fasulye ve mısır tarlalarını su bastı. (EN: to the center konaklı, edikli towns and karaatlı and potato, bean, and corn fields were flooded due to the hail and that was effective in.)
mBERT-cased	**niğde'de** dün akşam saatlerinde etkili olan sağanak nedeniyle 20 bin dekar **alanı** su bastı. (EN: 20 thousand decares of land were flooded in niğde due to the heavy rain that was effective yesterday evening.)

The example in Table 17 is an interesting one which showcases some of the challenges in agglutinative languages. Although the titles generated by the models are syntactically correct, their meanings are different. This is caused by the novel words that are introduced in the summaries. All the underlined texts in the table (except mBART) give the answer to the question *where* and result in different answers. Thus, slight changes to the morphemes of the words can alter the meaning of a whole sentence in Turkish. This is a factor that makes text generation more

https://doi.org/10.1017/S1351324922000195 Published online by Cambridge University Press

Table 17. An example from the test set of TR-News accompanied with the titles generated by the models. News article's abstract is given as the input, and the title of the article is expected as the output. The words in bold denote novel unigrams (unigrams which are not present in the input text) generated by the models, whereas the underlined texts are for reference in the discussion

Abstract	çoklu organ yetmezliğinden önceki gün hayatını kaybeden istanbul âşığı tarihçi semavi eyice, hocaların hocası halil inalcık'ın fatih haziresi'ndeki kabrinin yanı başına defnedilecek. (EN: historian semavi eyice, a lover of istanbul, has lost his life the earlier day due to multiple organ failure and will be buried next to the tomb of halil inalcık, the great teacher, in fatih cemetery.)
Reference (title)	tarihçi eyice, hocaların hocası inalcık'ın yanı başına defnedilecek (EN: historian eyice will be buried next to the tomb of inalcık, the great teacher)
mT5	istanbul âşığı eyice, halil inalcık'ın **kabrine** defnedilecek (EN: eyice, a lover of istanbul, will be buried to the tomb of halil inalcık)
mBART	semavi eyice **toprağa verilecek** (EN: semavi eyice will be buried)
BERTurk-uncased	semavi eyice, hocası halil inalcık ' ın kabrinin **yanında** defnedilecek (EN: semavi eyice will be buried at the side of his teacher halil inalcık's tomb)
BERTurk-cased	semavi eyice, hocası halil inalcık'ın **yanına** defnedilecek (EN: semavi eyice will be buried next to his teacher halil inalcık)
mBERT-uncased	halil inalcık'ın **hocasının yanındaki** kabrinin **yanına** defnedilecek (EN: will be buried next to the tomb that is next to the tomb of halil inalcık's teacher)
mBERT-cased	semavi eyice, halil inalcık ın yanı **başında** defnedilecek (EN: semavi eyice will be buried right beside halil inalcık)

challenging compared to languages such as English. In the case of mBART, it has provided a much less informative title and, however, has managed to produce the novel phrase "toprağa verilecek" (will be buried) which has the same meaning as the word "defnedilecek" in the abstract.

Table 18 shows another example for the title generation task. As in the previous example, all models except mBERT-uncased managed to produce syntactically correct titles. BERTurk-uncased, on the other hand, produced a semantically incorrect title by mistaking *Brexit* with *England*. The most abstractive output was generated by the mBERT-cased model producing novel unigrams and also generating the bi-gram "ingiltere başbakanı" (prime minister of England) which is not present in the abstract. However, it also failed to produce a meaningful title mistaking *Brexit* with the *EU*. Accordingly, the best titles that reflect the reference for this example belong to the mT5, mBART, and BERTurk-cased models.

7. Conclusion

In this paper, we analyzed in detail the performance of pretrained sequence-to-sequence models on two tasks, text summarization and title generation. The mT5 model reached the state-of-the-art results on both the TR-News and MLSum (TR) datasets in terms of the ROUGE scores for both tasks. The monolingual BERTurk-cased model also showed a performance close to the mT5 model and produced more novel summaries. We established strong baselines for both datasets for the summary generation task and also the title generation task for the Turkish language. Further analysis on the title generation task revealed that the input to the model impacts the task's outcome greatly. Providing the abstract of the news articles as input to the models showed better ROUGE scores compared to giving the LEAD sentences as input. Moreover, we created a larger dataset (Combined-TR) by combining both TR-News and MLSum (TR) since both have similar

https://doi.org/10.1017/S1351324922000195 Published online by Cambridge University Press

Table 18. An example from the test set of MLSum (TR) accompanied with the titles generated by the models. News article's abstract is given as the input, and the title of the article is expected as the output. The words in bold denote novel unigrams (unigrams which are not present in the input text) generated by the models

Abstract	ingiltere 'de resmen ülkenin yeni başbakanı olan boris johnson, 31 ekim'de brexit'i gerçekleştireceklerini, ve ab'den ayrılmaya hazır olduklarını açıkladı.
	(EN: boris hohnson, who is officially the new prime minister of the country in england, announced that they will hold brexit on october 31st and that they are ready to leave the eu.)
Reference (Title)	boris johnson'dan brexit mesajı! ab'den ayrılmaya hazırız
	(EN: brexit message from boris johnson! we are ready to leave the eu)
mT5	boris johnson ab'den ayrılmaya hazır
	(EN: boris johnson is ready to leave the eu)
mBART	ingiltere ab'den ayrılmaya hazır
	(EN: england is ready to leave the eu)
BERTurk-uncased	brexit : ab'den ayrılmaya **hazırız**
	(EN: brexit : we are ready to leave the eu)
BERTurk-cased	ingiltere ab'den ayrılmaya hazır
	(EN: england is ready to leave the eu)
mBERT-uncased	ingiltere **brexit'i hazırız**
	(EN: england ready brexit)
mBERT-cased	ingiltere başbakanı **brexit'ten ayrılıyor**
	(EN: prime minister of england is leaving brexit)

characteristics in terms of statistics and content. The models trained on Combined-TR showed performance gains for both the text summarization and title generation tasks. Lastly, the monolingual BERT models outperformed the multilingual BERT models in the BERT2BERT model architecture on both tasks.

In future works, we plan to extend this study with summarization datasets from different languages, specifically agglutinative languages. Given adequate computational resources, pretraining monolingual Seq2Seq models for low-resourced languages from scratch and comparing the results with the multilingual pretrained Seq2Seq models can be a future possibility. Moreover, the pretraining objectives can be altered to take into consideration the agglutinative nature of such languages.

References

Altan Z. (2004). A Turkish automatic text summarization system. In *IASTED International Conference on AIA*.

Baykara B. and Güngör T. (2022). Abstractive text summarization and new large-scale datasets for agglutinative languages Turkish and Hungarian. Language Resources and Evaluation, 1–35, ISSN 1574-020X.

Bostrom K. and Durrett G. (2020). Byte pair encoding is suboptimal for language model pretraining. In *Findings of the Association for Computational Linguistics: EMNLP 2020, Online, November 2020*. Association for Computational Linguistics, pp. 4617–4624.

Çelikyılmaz A., **Bosselut A.**, **He X. and Choi Y.** (2018). Deep communicating agents for abstractive summarization. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, New Orleans, Louisiana, June 2018. Association for Computational Linguistics, pp. 1662–1675.

Çığır C., **Kutlu M. and Çiçekli İ.** (2009). Generic text summarization for Turkish. In ISCIS. IEEE, pp. 224–229.

Chan B., **Schweter S. and Möller T.** (2020). German's next language model. CoRR, abs/2010.10906.

https://doi.org/10.1017/S1351324922000195 Published online by Cambridge University Press

Chopra S., **Auli M. and Rush A. M.** (2016). 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*, San Diego, California, June 2016. Association for Computational Linguistics, pp. 93–98.

Conneau A., **Khandelwal K.**, **Goyal N.**, **Chaudhary V.**, **Wenzek G.**, **Guzmán F.**, **Grave E.**, **Ott M.**, **Zettlemoyer L. and Stoyanov V.** (2020). Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Online, July 2020. Association for Computational Linguistics, pp. 8440–8451.

Devlin J., **Chang M.-W.**, **Lee K. and Toutanova K.** (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)*, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics, pp. 4171–4186.

Dong L., **Yang N.**, **Wang W.**, **Wei F.**, **Liu X.**, **Wang Y.**, **Gao J.**, **Zhou M. and Hon H.-W.** (2019). Unified language model pre-training for natural language understanding and generation. In Wallach H., Larochelle H., Beygelzimer A., d' Alché-Buc F., Fox E. and Garnett R. (eds), Advances in Neural Information Processing Systems, vol. 32. Curran Associates, Inc.

Edmundson H. P. (1969). New methods in automatic extracting. *Journal of ACM* **16**(2), 264–285,

Güran A., **Bayazit N.G. and Bekar E.** (2011). Automatic summarization of Turkish documents using non-negative matrix factorization. In *2011 International Symposium on Innovations in Intelligent Systems and Applications*. IEEE, pp. 480–484.

Güran A., **Bayazit N.G. and Gürbüz M.Z.** (2013). Efficient feature integration with Wikipedia-based semantic feature extraction for Turkish text summarization. *Turkish Journal of Electrical Engineering & Computer Sciences* **21**(5), 1411–1425.

Hermann K.M., **Kočiský T.**, **Grefenstette E.**, **Espeholt L.**, **Kay W.**, **Suleyman M. and Blunsom P.** (2015). Teaching machines to read and comprehend. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'15, Cambridge, MA, USA. MIT Press, pp. 1693–1701.

Hochreiter S. and Schmidhuber J. (1997). Long short-term memory. *Neural Computation* **9**(8), 1735–1780.

Hu J., **Ruder S.**, **Siddhant A.**, **Neubig G.**, **Firat O. and Johnson M.** (2020) XTREME: A massively multilingual multi-task benchmark for evaluating cross-lingual generalisation. In **Daume H.** III and Singh A. (eds), *Proceedings of the 37th International Conference on Machine Learning*, Proceedings of Machine Learning Research, vol. 119, 13–18 July 2020. PMLR, pp. 4411–4421.

Karakoç E. and Yılmaz B. (2019). Deep learning based abstractive Turkish news summarization. In *2019 27th Signal Processing and Communications Applications Conference (SIU)*, pp. 1–4.

Kartal Y.S. and Kutlu M. (2020). Machine learning based text summarization for Turkish news. In *2020 28th Signal Processing and Communications Applications Conference (SIU)*. IEEE, pp. 1–4.

Kingma D.P. and Ba J. (2015). Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations*, *ICLR 2015, San Diego, CA, USA, 7–9 May 2015, Conference Track Proceedings*.

Kudo T. (2018). Subword regularization: Improving neural network translation models with multiple subword candidates. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume* 1: *Long Papers)*, Melbourne, Australia, July 2018. Association for Computational Linguistics, pp. 66–75.

Kuratov Y. and Arkhipov M. (2019). Adaptation of Deep Bidirectional Multilingual Transformers for Russian Language. CoRR, abs/1905.07213.

Lewis M., **Liu Y.**, **Goyal N.**, **Ghazvininejad M.**, **Mohamed A.**, **Levy O.**, **Stoyanov V. and Zettlemoyer L.** (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*, Online, July 2020. Association for Computational Linguistics, pp. 7871–7880.

Lin C.-Y. (2004). ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, Barcelona, Spain, July 2004. Association for Computational Linguistics, pp. 74–81.

Liu Y. and Lapata M. (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)*, Hong Kong, China, November 2019. Association for Computational Linguistics, pp. 3730–3740.

Liu Y., **Gu J.**, **Goyal N.**, **Li X.**, **Edunov S.**, **Ghazvininejad M.**, **Lewis M. and Zettlemoyer L.** (2020). Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics* **8**, 726–742.

Liu Y., **Ott M.**, **Goyal N.**, **Du J.**, **Joshi M.**, **Chen D.**, **Levy O.**, **Lewis M.**, **Zettlemoyer L. and Stoyanov V.** (2019). Roberta: A Robustly Optimized BERT Pretraining Approach. CoRR, abs/1907.11692.

Luhn H.P. (1958). The automatic creation of literature abstracts. *IBM Journal of Research and Development* **2**(2), 159–165.

Mihalcea R. and Tarau P. (2004). TextRank: Bringing order into text. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, Barcelona, Spain, July 2004. Association for Computational Linguistics, pp. 404–411.

Nallapati R., **Zhou B.**, **dos Santos C.**, **Gülçehre C. and Xiang B.** (2016). Abstractive text summarization using sequence-to-sequence RNNs and beyond. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, Berlin, Germany, August 2016. Association for Computational Linguistics, pp. 280–290.

Nallapati R., Zhai F. and Zhou B. (2017). SummaRuNNer: A recurrent neural network based sequence model for extractive summarization of documents. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17. AAAI Press, pp. 3075–3081.

Narayan S., Cohen S.B. and Lapata M. (2018a). Don't give me the details, just the summary! Topic-aware convolutional neural networks for extreme summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, Belgium, October–November 2018. Association for Computational Linguistics, pp. 1797–1807.

Narayan S., Cohen S.B. and Lapata M. (2018b). 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)*, New Orleans, Louisiana, June 2018. Association for Computational Linguistics, pp. 1747–1759.

Ng A., Ngiam J., Foo C.Y. and Mai Y. (2014). Deep learning. In *CS229 Lecture Notes*, pp. 1–30.

Oflazer K. (2014). Turkish and its challenges for language processing. *Language Resources and Evaluation* **48**(4), 639–653.

Özsoy M.G., Çiçekli İ. and Alpaslan F.N. (2010). Text summarization of Turkish texts using latent semantic analysis. In *Proceedings of the 23rd International Conference on Computational Linguistics*, COLING'10, USA. Association for Computational Linguistics, pp. 869–876.

Paulus R., Xiong C. and Socher R. (2018). A deep reinforced model for abstractive summarization. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30–May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Pembe F.C. and Güngör T. (2008). Towards a new summarization approach for search engine results: An application for Turkish. In *2008 23rd International Symposium on Computer and Information Sciences*. IEEE, pp. 1–6.

Polignano M., Basile P., de Gemmis M., Semeraro G. and Basile V. (2019). AlBERTo: Italian BERT language understanding model for NLP challenging tasks based on Tweets. In *Proceedings of the Sixth Italian Conference on Computational Linguistics (CLiC-it 2019)*, vol. 2481. CEUR.

Qi W., Yan Y., Gong Y., Liu D., Duan N., Chen J., Zhang R. and Zhou M. (2020). Prophetnet: Predicting future n-gram for sequence-to-sequence pre-training. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pp. 2401–2410.

Radford A., Narasimhan K., Salimans T. and Sutskever I. (2018). Improving language understanding by generative pre-training.

Raffel C., Shazeer N., Roberts A., Lee K., Narang S., Matena M., Zhou Y., Li W. and Liu P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research* **21**(140), 1–67.

Rajpurkar P., Zhang J., Lopyrev K. and Liang P. (2016). SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, Austin, Texas, November 2016. Association for Computational Linguistics, pp. 2383–2392.

Rothe S., Narayan S. and Severyn A. (2020). Leveraging pre-trained checkpoints for sequence generation tasks. *Transactions of the Association for Computational Linguistics* **8**, 264–280.

Rush A.M., Chopra S. and Weston J. (2015). A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, Lisbon, Portugal, September 2015. Association for Computational Linguistics, pp. 379–389.

Rust P., Pfeiffer J., Vulic I., Ruder S. and Gurevych I. (2020). How Good is Your Tokenizer? On the Monolingual Performance of Multilingual Language Models. CoRR, abs/2012.15613.

Schweter S. (2020) *BERTurk - BERT models for Turkish*, April 2020. URL https://doi.org/10.5281/zenodo.3770924.

Scialom T., Dray P.-A., Lamprier S., Piwowarski B. and Staiano J. (2020). MLSUM: The multilingual summarization corpus. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Online, November 2020. Association for Computational Linguistics, pp. 8051–8067.

See A., Liu P.J. and **Manning C.D.** (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)*, Vancouver, Canada, July 2017. Association for Computational Linguistics, pp. 1073–1083.

Sennrich R., Haddow B. and Birch A. (2016). Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume* 1: *Long Papers)*, Berlin, Germany, August 2016. Association for Computational Linguistics, pp. 1715–1725.

Shazeer N. and Stern M. (2018). Adafactor: Adaptive learning rates with sublinear memory cost. In **Dy J. and Krause A.** (eds), *Proceedings of the 35th International Conference on Machine Learning*, Proceedings of Machine Learning Research, vol. 80, 10–15 July 2018. PMLR, pp. 4596–4604.

Song K., Tan X., Qin T., Lu J. and Liu T.-Y. (2019). MASS: Masked sequence to sequence pre-training for language generation. In International Conference on Machine Learning, pp. 5926–5936.

Torres-Moreno J.-M. (2014). *Automatic Text Summarization*. London: John Wiley & Sons.

Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A.N., Kaiser L.U. and Polosukhin I. (2017). Attention is all you need. *In Advances in Neural Information Processing Systems*, vol. **30**. Curran Associates, Inc.

Virtanen A., Kanerva J., Ilo R., Luoma J., Luotolahti J., Salakoski T., Ginter F. and Pyysalo S. (2019). Multilingual is not enough: BERT for Finnish. CoRR, abs/1912.07076.

https://doi.org/10.1017/S1351324922000195 Published online by Cambridge University Press

Wenzek G., **Lachaux M.-A.**, **Conneau V**, **Chaudhary V.**, **Guzmán F.**, **Joulin A. and Grave E.** (2020). CCNet: Extracting high quality monolingual datasets from web crawl data. In *Proceedings of the 12th Language Resources and Evaluation Conference*, Marseille, France, May 2020. European Language Resources Association, pp. 4003–4012.

Wolf T., **Debut L.**, **Sanh V.**, **Chaumond J.**, **Delangue C.**, **Moi A.**, **Cistac P.**, **Rault T.**, **Louf R.**, **Funtowicz M.**, **Davison J.**, **Shleifer S.**, **von Platen P.**, **Ma C.**, **Jernite Y.**, **Plu J.**, **Xu C.**, **Le Scao T.**, **Gugger S.**, **Drame M.**, **Lhoest Q. and Rush A.** (2020). Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, Online, October 2020. Association for Computational Linguistics, pp. 38–45.

Wu Y., **Schuster M.**, **Chen Z.**, **Le Q.V.**, **Norouzi M.**, **Macherey W.**, **Krikun M.**, **Cao Y.**, **Gao Q.**, **Macherey K.**, **Klingner J.**, **Shah A.**, **Johnson M.**, **Liu X.**, **Łukasz Kaiser**, **Gouws S.**, **Kato Y.**, **Kudo T.**, **Kazawa H.**, **Stevens K.**, **Kurian G.**, **Patil N.**, **Wang W.**, **Young C.**, **Smith J.**, **Riesa J.**, **Rudnick A.**, **Vinyals O.**, **Corrado G.**, **Hughes M. and Dean J.** (2016). Google's neural machine translation system: Bridging the gap between human and machine translation. CoRR, abs/1609.08144.

Xue L., **Constant N.**, **Roberts A.**, **Kale M.**, **Al-Rfou R.**, **Siddhant A.**, **Barua A. and Raffel C.** (2021). mt5: A massively multilingual pre-trained text-to-text transformer. ArXiv, abs/2010.11934.

Yang Z., **Dai Z.**, **Yang Y.**, **Carbonell J.**, **Salakhutdinov R.R. and Le Q.V.** (2019). XLNet: Generalized autoregressive pretraining for language understanding. In Wallach H., Larochelle H., Beygelzimer A., d' Alché-Buc F., Fox E. and Garnett R. (eds), Advances in Neural Information Processing Systems, vol. 32. Curran Associates, Inc.,

Zhang J., **Zhao Y.**, **Saleh M. and Liu P.** (2020). PEGASUS: Pre-training with extracted gap-sentences for abstractive summarization. In Daume H. III and Singh A. (eds), *Proceedings of the 37th International Conference on Machine Learning*, Proceedings of Machine Learning Research, vol. 119, 13–18 July 2020. PMLR, pp. 11328–11339.

Appendix

Table A1. Cross-dataset evaluation results for the summary generation task

	TR-News			MLSum (TR)			Combined-TR		
Model & training set	R1	R2	RL	R1	R2	RL	R1	R2	RL
mT5-TR-News	41.13	25.75	37.6	40.99	26.24	36.77	41.06	25.97	37.22
mT5-MLSum-TR	37.25	22.1	33.66	42.26	27.81	37.96	39.52	24.69	35.61
mT5-Combined-TR	41.23	25.98	37.73	44.01	29.49	39.79	42.49	27.58	38.67

Table A2. Cross-dataset evaluation results for the title generation (abstract as input) task

	TR-News			MLSum (TR)			Combined-TR		
Model & training set	R1	R2	RL	R1	R2	RL	R1	R2	RL
mT5-TR-News	41.87	24.49	40.87	41.81	23.08	39.87	41.84	23.87	40.41
mT5-MLSum-TR	36.32	19.05	35.3	40.77	22.42	38.97	38.34	20.59	36.97
mT5-Combined-TR	42.46	24.96	41.41	43.79	25.32	41.81	43.04	25.14	41.59

https://doi.org/10.1017/S1351324922000195 Published online by Cambridge University Press

Table A3. The analysis results for the summary generation task given various beam sizes and early-stopping method

	TR-News			MLSum (TR)			Combined-TR		
Parameters	R1	R2	RL	R1	R2	RL	R1	R2	RL
mT5-beam-1	40.74	24.98	37.3	40.87	25.83	36.65	41.99	26.51	38.26
mT5-beam-2	41.34	25.81	37.9	42.13	27.44	37.85	42.61	27.48	38.84
mT5-beam-3	41.3	25.87	37.8	42.18	27.66	37.92	42.59	27.62	38.82
mT5-beam-4	41.13	25.75	37.6	42.26	27.81	37.96	42.49	27.58	38.67
mT5-beam-4 & early-stopping	41.15	25.74	37.86	41.36	26.92	37.32	42.18	27.22	38.61

Table A4. The analysis results for the title generation (abstract as input) task given various beam sizes and early-stopping method

	TR-News			MLSum (TR)			Combined-TR		
Parameters	R1	R2	RL	R1	R2	RL	R1	R2	RL
mT5-beam-1	40.41	22.76	39.41	37.93	19.81	36.25	41.12	23.11	39.75
mT5-beam-2	41.58	24.1	40.56	39.95	21.65	39.19	42.49	24.51	41.08
mT5-beam-3	41.82	24.39	40.81	40.54	22.22	38.76	42.91	25	41.49
mT5-beam-4	41.87	24.49	40.87	40.77	22.42	38.97	43.04	25.14	41.59
mT5-beam-4 & early-stopping	41.66	24.17	40.73	40.04	21.7	38.3	42.53	24.57	41.18

Table A5. ROUGE scores with different preprocessing settings for the summary generation task. “Punct removed” refers to removing the punctuations, whereas “Punct kept” refers to keeping the punctuations before the ROUGE calculations. “Stems taken” refers to applying stemming operation on the words, whereas “Stems not taken” refers to leaving the words in their surface form before the ROUGE calculations

	TR-News			MLSum (TR)			Combined-TR		
Parameters	R1	R2	RL	R1	R2	RL	R1	R2	RL
Punct removed & stems taken	41.13	25.75	37.60	42.26	27.81	37.96	42.49	27.58	38.67
Punct removed & stems not taken	37.60	23.93	34.89	39.03	26.22	35.57	39.12	25.85	36.12
Punct kept & stems taken	43.64	25.75	39.66	44.60	27.67	39.90	44.83	27.46	40.59
Punct kept & stems not taken	40.55	24.17	37.34	41.76	26.29	37.86	41.88	25.94	38.41

Table A6. ROUGE scores with different preprocessing settings for the title generation (abstract as input) task. “Punct removed” refers to removing the punctuations, whereas “Punct kept” refers to keeping the punctuations before the ROUGE calculations. “Stems taken” refers to applying stemming operation on the words, whereas “Stems not taken” refers to leaving the words in their surface form before the ROUGE calculations

	TR-News			MLSum (TR)			Combined-TR		
Parameters	R1	R2	RL	R1	R2	RL	R1	R2	RL
Punct removed & stems taken	41.87	24.49	40.87	40.77	22.42	38.97	43.04	25.14	41.59
Punct removed & stems not taken	37.91	22.30	37.15	37.22	20.65	35.79	39.24	23.05	38.10
Punct kept & stems taken	40.00	23.02	39.02	39.23	20.79	37.37	41.09	23.44	39.70
Punct kept & stems not taken	36.35	21.07	35.60	35.92	19.20	34.42	37.56	21.57	36.47

https://doi.org/10.1017/S1351324922000195 Published online by Cambridge University Press

Table A7. ROUGE-1 scores of all the models calculated under different preprocessing settings on the TR-News dataset for the text summarization task. “Punct removed” refers to removing the punctuations, whereas “Punct kept” refers to keeping the punctuations before the ROUGE calculations. “Stems taken” refers to applying stemming operation on the words, whereas “Stems not taken” refers to leaving the words in their surface form before the ROUGE calculations

	TR-News text summarization task					
Parameters	mT5	BERTurk-uncased	BERTurk-cased	mBART	mBERT-uncased	mBERT-cased
Punct removed & stems taken	**41.13**	40.50	41.06	40.52	33.04	39.73
Punct removed & stems not taken	37.60	37.13	**37.63**	36.97	30.38	36.22
Punct kept & stems taken	**43.64**	42.34	42.85	43.05	35.78	41.37
Punct kept & stems not taken	**40.55**	39.88	39.43	39.95	33.52	38.30

Table A8. ROUGE-1 scores of all the models calculated under different preprocessing settings on the MLSum (TR) dataset for the text summarization task. “Punct removed” refers to removing the punctuations, whereas “Punct kept” refers to keeping the punctuations before the ROUGE calculations. “Stems taken” refers to applying stemming operation on the words, whereas “Stems not taken” refers to leaving the words in their surface form before the ROUGE calculations

	MLSum (TR) text summarization task					
Parameters	mT5	BERTurk-uncased	BERTurk-cased	mBART	mBERT-uncased	mBERT-cased
Punct removed & stems taken	**42.26**	41.47	41.48	40.47	33.59	40.27
Punct removed & stems not taken	**39.03**	38.35	38.40	37.27	31.27	37.16
Punct kept & stems taken	**44.60**	43.33	43.28	42.95	41.89	36.28
Punct kept & stems not taken	**41.76**	40.62	40.59	40.14	34.28	39.15

Table A9. ROUGE-1 scores of all the models calculated under different preprocessing settings on the Combined-TR dataset for the text summarization task. “Punct removed” refers to removing the punctuations, whereas “Punct kept” refers to keeping the punctuations before the ROUGE calculations. “Stems taken” refers to applying stemming operation on the words, whereas “Stems not taken” refers to leaving the words in their surface form before the ROUGE calculations

	Combined-TR text summarization task					
Parameters	mT5	BERTurk-uncased	BERTurk-cased	mBART	mBERT-uncased	mBERT-cased
Punct removed & stems taken	42.49	42.51	**42.75**	41.97	34.13	41.20
Punct removed & stems not taken	39.12	39.20	**39.47**	38.56	31.60	37.82
Punct kept & stems taken	**44.83**	44.14	44.33	44.32	36.70	42.69
Punct kept & stems not taken	**41.88**	41.28	41.48	41.34	34.55	39.73

Cite this article: Baykara B and Güngör T (2023). Turkish abstractive text summarization using pretrained sequence-to-sequence models. *Natural Language Engineering* **29**, 1275–1304. https://doi.org/10.1017/S1351324922000195