Annales Mathematicae et Informaticae
53 (2021) pp. 299–316
DOI: https://doi.org/10.33039/ami.2021.04.002
URL: https://ami.uni-eszterhazy.hu

Abstractive text summarization for Hungarian

Zijian Győző Yang[abc], Ádám Agócs[a], Gábor Kusper[a], Tamás Váradi[c]

[a]Eszterházy Károly University, Faculty of Informatics
agadam98@gmail.com
{yang.zijian.gyozo,kusper.gabor}@uni-eszterhazy.hu

[b]MTA-PPKE Hungarian Language Technology Research Group
yang.zijian.gyozo@itk.ppke.hu

[c]Hungarian Research Centre for Linguistics
{yang.zijian.gyozo,varadi.tamas}@nytud.hu

Submitted: November 22, 2020
Accepted: April 12, 2021
Published online: May 18, 2021

Abstract

In our research we have created a text summarization software tool for Hungarian using multilingual and Hungarian BERT-based models. Two types of text summarization method exist: abstractive and extractive. The abstractive summarization is more similar to human generated summarization. Target summaries may include phrases that the original text does not necessarily contain. This method generates the summarized text by applying keywords that were extracted from the original text. The extractive method summarizes the text by using the most important extracted phrases or sentences from the original text. In our research we have built both abstractive and extractive models for Hungarian. For abstractive models, we have used a multilingual BERT model and Hungarian monolingual BERT models. For extractive summarization, in addition to the BERT models, we have also made experiments with ELECTRA models. We find that the Hungarian monolingual models outperformed the multilingual BERT model in all cases. Furthermore, the ELECTRA small models achieved higher results than some of the BERT models. This result is important because the ELECTRA small models have much fewer parameters and were trained on only 1 GPU within a couple of days. Another important consideration is that the ELECTRA models are much smaller than the BERT models, which is important for the end users. To our best knowledge the first extractive and abstractive summarization systems reported in the present paper are the first such systems for Hungarian.

Keywords: BERT, huBERT, ELECTRA, HILBERT, abstractive summarization, extractive summarization

AMS Subject Classification: 68T07, 68T50, 68T09

1. Introduction

Processing large amounts of textual data in our everyday life with manual tools is proving increasingly difficult because of the scale of the data. For instance, any company or public institution typically has an enormous amount of text data. It may be especially important for them to extract the essence of data from the huge body of texts. Using automatic methods for extracting or summarizing can lead to significant saving of time and costs. Therefore, there is an increasing demand for automatic information extraction applications. Automatic text summarization is a particularly pressing, unsolved challenge for the Hungarian language.

Automatic text summarization is the process of shortening a text document using a system for prioritizing information. Technologies that generate summaries take into account variables such as length, style, or syntax. Text summarization from the perspective of humans is taking a chunk of information and extracting the most important parts from it. Automatic text summarization methods typically rely on the logical quantification of features of the text including weighting keywords, and sentence ranking.

There are two different machine summarization methods: extractive and abstractive summarization.

Abstractive text summarization can generate completely new pieces of texts while capturing the meaning of the original article. Abstractive methods are usually more complex because the machine has to analyze the text and the most important information from it, then learn the relevant concepts and construct cohesive summaries.

Extractive text summarization does not generate any new text, it only uses words already in the original article and combines the existing words, phrases or sentences that are the most relevant to the article. Extractive summarization techniques include ranking sentences and phrases in order of importance, and selecting the most important components of a document to create a summary.

In our research we have carried out both extractive and abstractive experiments for Hungarian.

2. Related work

The extractive method creates summarization by selecting the most important phrases or sentences from the original text. It involves a classification problem:

the task is to find which sentences should be selected for inclusion in the summary. One of the first neural network-based extractive summarization tool is SummaRuNNer [13], which uses an RNN encoder to solve the problem. Another method called Refresh [14] is based on the Rouge metric, which is used to rank sentences in the text using the reinforcement learning method. The goal of Latent [25] was to propose a latent variable extractive model where sentences are viewed as latent variables and sentences with activated variables are used to infer gold summaries. Sumo [9] uses a method that builds on multi-root dependency tree structures that can be extracted from a document and predicts the possible form of the summary. NeuSum [26] approaches the problem by scoring and selecting sentences from the original text.

The abstractive summarization with neural network approaches the problem as a transformation from a sequence into another sequence. The encoder identifies tokens from the source document, then maps them onto target tokens, and finally generates new text from the decoder. The PTgen [19] tool generates pointers to identify words in the source text, then using a coverage mechanism keeps the words to generate the summary. Deep Communicating Agent [1] is an agent-based approach where the task of encoding a long text is shared among multiple collaborating agents, each in charge of a subsection of the input text. These encoders are connected to a single decoder, trained end-to-end using reinforcement learning to generate a focused and coherent summary. The Deep Reinforced Model [17] uses an intra-attention that attends to the input and the continuously generated output separately, as well as a new training method that combines standard supervised word prediction and reinforcement learning. The Bottom-Up [4] approach uses a data-efficient content selector to "over-determine" phrases in a source document that should be part of the summary. The method uses this selector as a bottom-up attention step to constrain the model to likely phrases.

The PreSumm [8] model was the state-of-the-art tool in 2019. It requires a pretrained BERT model to train extractive and abstractive summarization models. Pre-training a BERT model requires huge data and compute capacity. Fortunately, we can choose the PreSumm model because recently a number of BERT models have been created for Hungarian. We can use the multilingual BERT[1], which, of course, covers Hungarian. There are also two Hungarian monolingual BERT base models built by Nemeskey [16] that we could use for our research.

In the recent months, further models for Hungarian were successfully trained[2]: HIL-ELECTRA, HIL-RoBERTa, HIL-ALBERT and HILBERT [3]. For the purposes of the present research we experimented with the HIL-ELECTRA and the HILBERT models.

In recent months, autoregressive methods achieved the best results in the field of summarization. Autoregressive models rely on the decoder of the transformer model and use an attention mask on the top of the full sentence so the model can only look at the tokens before the current text. This method achieved higher results

[1] `https://github.com/google-research/bert/blob/master/multilingual.md`

[2] `https://hilanco.github.io/`

on many text generation tasks [22]. The BART model[6] is a denoising autoencoder for pretraining sequence-to-sequence models, which is trained to corrupt text with arbitrary noising function and then to learn to reconstruct the original text. This model is effective for fine-tuning summarization tasks.

Currently, the state-of-the-art tool for summarization is the PEGASUS [24] system. In PEGASUS, important sentences are removed/masked from an input document and are generated together as one output sequence from the remaining sentences, similar to an extractive summary. For Hungarian, the OpinHu system has a summary function [10]. The system uses keywords and text context to extract information. Lengyelné Molnár Tünde [12] examined the possibilities and limitations of the automatic generation of research abstracts. Using the PreSumm [8] tool, Yang et al. built the first extractive summarization tool [23] for Hungarian. In this paper we present the first Hungarian abstractive summarization tool. It was built using the PreSumm system.

3. BERT and ELECTRA models

In our experiments, different kinds of BERT models were used for our summarization tasks, ELECTRA model was tested in addition to the BERT model for the extractive summarization.

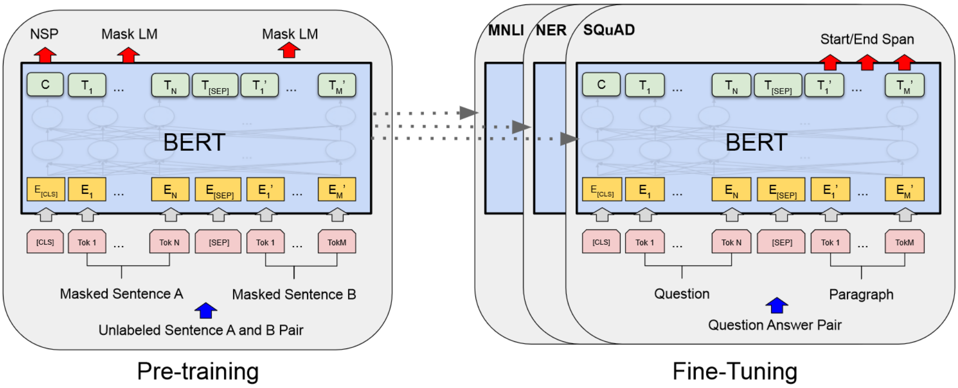

Figure 1. BERT model.

BERT (Bidirectional Encoder Representations from Transformer) is a multi-layer, bidirectional Transformer encoder [21]. The BERT model was trained on two language modeling tasks (see Figure 1): masking and next sentence prediction. During masking, 15% of the words in the corpus are randomly masked, then the system had to learn the correct word. During the next sentence prediction task, the model receives two sentences, the task is to guess whether the two received sentences are next to each other in the original text or two randomly selected sentences. To limit the size of the dictionary and to solve the out-of-vocabulary words problem, WordPiece tokenizer [18] was used. After training BERT, the pretrained model is used to fine-tune to the target task. The BERT model is further trained on a specific downstream task with a feed-forward network in the fine-tuning process.

One of the advantages of BERT is that models have not only been trained on English. Google has trained two multilingual models[3]: lowercase and non-lowercase. The first 104 languages with the largest Wikipedia were selected to train the models. The size of Wikipedia varies greatly between the languages, the English Wikipedia accounting for nearly 20% of the data, so sampling was controlled by normalization to solve this problem. Then, all languages, same like English, were tokenized, which had four steps: lowercasing, accent removal, punctuation and whitespace handling. Training the non-lowercase model also followed these steps except lowercasing. WordPiece tokenization and dictionary can handle cased and unknown words. The Hungarian language is also part of this model.

The first Hungarian BERT model was published by Dávid Márk Nemeskey [16], which is called huBERT[4]. Three huBERT models were trained:

- huBERT: BERT base model trained on Hungarian Webcorpus 2.0[5]
- huBERT Wikipedia cased: cased BERT base model trained on Hungarian Wikipedia
- huBERT Wikipedia lowercased: lowercased BERT base model trained on Hungarian Wikipedia

Currently the huBERT models achieve state-of-the-art results in name entity recognition and noun phrase chunking tasks [15].

ELECTRA [2] is based on the GAN (Generative adversarial network) [5] method. The basis of the method (see Figure 2) is that there are two networks are trained, a generator and a discriminator. During training, the generator randomly generates vector representations from which it generates output. Then, real output data is shown, which can improve the performance of random vector generation. In this way, by the end of the training, the generator will become "smarter" and will be able to generate an output that closely matches the real output. The discriminator is trained to predict whether a particular word is the original word or a replacement. During training, the discriminator gets data from the real corpus, and also gets data that generated by the generator. By the end, the generator can generate content that similar to a real content, the discriminator can distinguish between a fake/erroneous content and a real/correct content.

ELECTRA is a modified GAN method for training language model (see Figure 2). The difference compared to the BERT model (and the original GAN) is that ELECTRA does not try to predict the original word behind the masked word but instead the generator randomly generates words for the masked words and then the discriminator has to guess if the words given by the generator are the original words

[3] `https://github.com/google-research/bert/blob/master/multilingual.md`

[4] `https://hlt.bme.hu/en/resources/hubert`

[5] `https://hlt.bme.hu/en/resources/webcorpus2`

or randomly generated words. Thus, the generator slowly learns what actual words match to the place of the masked words, while the discriminator learns whether the given input text are built with real words or fake words. After training, the generator is discarded and only the discriminator is retained for fine-tuning.

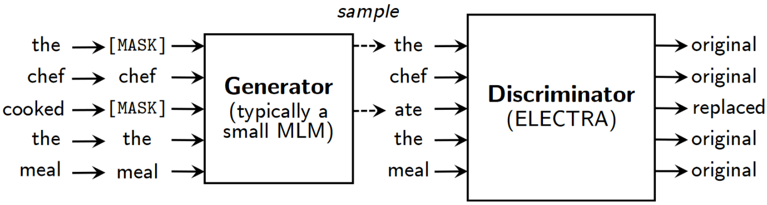

Figure 2. ELECTRA model.

4. Corpora

For building the summarization corpora for fine-tuning, we used 3 different kinds of resource: HVG, index.hu and the Hungarian MARCELL corpus [20]. Table 1 displays the main characteristics of the corpora.

Table 1. Main characteristics of the corpora.

	HVG	index.hu	H+I	MARCELL
year	2012 - 2020	1999 - 2020	-	1991 - 2019
documents	480,660	183,942	559,162	24,747
token	129,833,741	104,640,902	159,131,373	28,112,090
type	5,133,030	3,921,893	3,053,703	450,115
avg tokens in src	246,27	496,27	265,17	1124,82
avg tokens in tgt	12,43	22,33	29,97	11,22
avg sents in src	23,74	35,76	11,40	49,26
avg sents in tgt	1,46	2,23	1,57	1,00

In the case of HVG[6] and index.hu[7], the body of the articles taken from the daily online newspaper, as well as the corresponding leads, representing the summaries. We have built two corpora from them. In the first version, we used only the HVG documents. In the second version (H+I corpus) we merged the HVG and the index.hu articles. In the case of MARCELL, we used the legal documents as source and each of these have one short sentence topic description that we used for target summary.

A BERT model has a maximum 512 sequence length (after BERT subword tokenization). Therefore in our research we used only the online daily articles and

[6] https://www.hvg.hu

[7] https://www.index.hu

its leads, because the articles of the weekly newspaper (HVG) are much longer. In the case of MARCELL, the average sentence length is 1124,82, which is much longer than 512, but the median is: 340, which is short enough for this task.

We did three different tasks. In the first two task, we used the HVG and MARCELL corpora on their own, without any cleaning and normalizing processes. In the third task, we merged the HVG and the index.hu corpora, and we also made cleaning processes on it. The cleaning and normalizing aspects are as followed:

- removed the long (500< token) documents from the corpora
- removed the short (5> token) documents from the corpora
- removed documents, that articles were shorter than its' lead (e.g., See Table 7)
- removed irrelevant articles or text parts: e.g. "Follow us on facebook", "Edited: [NAME]", "Click for more details", "Start a Quiz", etc.

5. Pretrained language models

In our abstractive summarization experiments we used 4 different kinds of pretrained BERT models: huBERT, huBERT Wikipedia cased, HILBERT, BERT-Base-Multilingual-Cased.

huBERT [15] is the state-of-the-art Hungarian cased (not lowercased) BERT-base model that trained on Webcorpus 2.0[8] (9 billion token) with 110 million parameters, 12-layer, 768-hidden, 12-heads.

huBERT Wikipedia cased [15] is a Hungarian cased BERT-base model that trained on Hungarian Wikipedia (170 million token) with 110 million parameters, 12-layer, 768-hidden, 12-heads.

HILBERT [3] is a Hungarian cased BERT-large model that trained on NYTK v1 corpus (3.7 billion token) with 340 million parameters, 24-layer, 1024-hidden, 16-heads.

BERT-Base-Multilingual-Cased[9] is a cased BERT-base model that trained on 104 languages of Wikipedia, with 110 million parameters, 12-layer, 768-hidden, 12-heads.

In the extractive summarization experiments, we used the 2 kinds of huBERT, the multilingual and 4 kinds of ELECTRA models. In the case of ELECTRA, there were no pretrained models for Hungarian, thus we did experiments to create them.

For training ELECTRA models, we have used three different corpora:

[8] `https://hlt.bme.hu/en/resources/webcorpus2`

[9] `https://github.com/google-research/bert/blob/master/multilingual.md`

- Hungarian Wikipedia (wiki): 13,098,808 segments; 163,772,783 tokens;
- NYTK corpus (NYTK): 283,099,534 segments; 3,993,873,992 tokens; (contains Hungarian Wikipedia)

The vocabulary size was 64,000. This relatively large size was deemed justified in view of the agglutinative nature of the rich morphological system of Hungarian resulting in an almost open-ended stock of wordforms. The ratio of number of subword tokens per surface words was 1.15707, which can be considered good.

To train ELECTRA models, we used the code[10] published by Google. We trained six different ELECTRA models for Hungarian, which we named as HIL-ELECTRA (HILANCO[11] ELECTRA):

- HIL-ELECTRA small wiki: trained on Hungarian Wikipedia. Training time: ~5 days
- HIL-ELECTRA small NYTK: trained on Hungarian Research Centre for Linguistics corpus v1. Training time: ~7 days
- HIL-ELECTRA base wiki: trained on Hungarian Wikipedia. Training time: ~5 days
- HIL-ELECTRA base NYTK: trained on Hungarian Research Centre for Linguistics corpus v1 corpus. Training time: ~7 days

In Table 2, we can see the training hyper-parameters of the ELECTRA small and base models.

Table 2. The hyper-parameters of the training of the ELECTRA models.

	Learning rate	Weight decay	Layers	Embedding size	Batch size	Training step
small	5e-4	0.01	12	128	80	1 million
base	5e-4	0.01	12	768	2	1 million

Each model was trained on 1 single GeForce RTX 2080 Ti type video card. The training took about 5-7 days. The run time is also affected by dictionary size, it can be accelerated with a smaller dictionary.

6. Experiments

Using the pretrained language models, we fine-tuned summarization models for Hungarian. The first step of our research was to pre-process the original text.

[10] `https://github.com/google-research/electra`

[11] `https://hilanco.github.io`

The articles and their leads were tokenized with the e-magyar[12] tokenizer module, the quntoken [11] tool. Then, we converted the tokenized text to JSON format for the summarization system. The system then inserts two special elements, the first one indicating the beginning of the text and the other one marks the sentence boundaries. After pre-processing, we trained different summarization models.

We trained and compared the following models in the different tasks:

- Abstractive summarization:
  - BERT Base Multilingual Cased (multi-BERT)
  - huBERT Wikipedia cased (huBERT wiki)
  - huBERT (huBERT web)
  - HILBERT
- Extractive summarization:
  - BERT Base Multilingual Cased
  - huBERT Wikipedia cased
  - huBERT
  - HIL-ELECTRA base Hungarian Wikipedia (HIL-ELECTRA base wiki)
  - HIL-ELECTRA base NYTK (HIL-ELECTRA base NYTK)
  - HIL-ELECTRA small Hungarian Wikipedia (HIL-ELECTRA small wiki)
  - HIL-ELECTRA small NYTK (HIL-ELECTRA small NYTK)

To train abstractive and extractive models, we used the PreSumm [8] tool[13]. In the Table 3 you can see the differences of BERT-base and BERT-large training (fine-tuning) hyper-parameters and characteristics. All other hyperparameters were set to default of experiments of Yang et al. [8].

Table 3. Differences of BERT-base and BERT-large training hyper-parameters.

	learning rate	lr decrease	batch size	hardware
BERT-base	1e-03	0.1	20	4x GeForce RTX 2080
BERT-large	5e-05	0.02	10	4x Tesla V100

In our experiments, the larger the corpus the more training steps are required. Accordingly, the following training steps were used:

- Abstractive summarization
  - MARCELL: 50,000

[12] `https://e-magyar.hu`

[13] `https://github.com/nlpyang/PreSumm`

  - HVG: 200,000
  - H+I: multi and huBERT: 600,000; HILBERT: 800,000

- Extractive summarization: 50,000

7. Results and Evaluation

The ROUGE [7] method was used for evaluation. ROUGE (Recall-Oriented Understudy for Gisting Evaluation) is a coverage-based method based on BLEU metrics used in machine translation. ROUGE itself contains several methods, of which ROUGE-1, ROUGE-2 and ROUGE-L methods were used for our measurements. ROUGE-1 is a unigram, while ROUGE-2 is a bigram coverage calculation algorithm. ROUGE-L examines the longest common word sequence at the paragraph and sentence level.

In Table 4, 5 and 6, we can see the ROUGErecall results of our abstractive and extractive experiments. Since the HILBERT model needs huge amount of resources, we used it only in the experiment of H+I and in this task we did not use the huBERT wiki, because the huBERT web contains the wiki.

Table 4. ROUGE recall results of abstractive summarization of MARCELL, HVG and H+I tasks.

		ROUGE-1	ROUGE-2	ROUGE-L
MARCELL	multi	87.37	77.38	84.97
	huBERT wiki	89.37	79.91	86.14
	huBERT web	**89.64**	**80.29**	**86.46**
HVG	multi	47.02	19.72	39.29
	huBERT wiki	49.49	21.62	41.46
	huBERT web	**51.47**	**23.27**	**43.82**
H+I	multi (550k)	51.85	23.22	43.45
	huBERT web (450k)	**57.07**	**26.97**	**48.28**
	HILBERT (800k)	44.98	14.22	37.06

Table 5. ROUGE F1 results of the first generated sentence of MARCELL tasks.

	ROUGE-1	ROUGE-2	ROUGE-L
multi	72.99	65.38	71.53
huBERT wiki	74.23	66.56	72.90
huBERT web	**75.85**	**68.35**	**74.61**

In Table 4, you can see only the recall results, because of the number of generated sentences are more than the reference sentences, the precision of ROUGE cannot show evaluable performance. In the case of MARCELL task, the result

Table 6. ROUGE recall results of extractive summarization.

	ROUGE-1	ROUGE-2	ROUGE-L
multi-BERT	48.58	20.12	39.42
huBERT wiki	48.86	20.45	39.60
huBERT web	**49.45**	**21.07**	**40.14**
ELECTRA base wiki	48.83	20.37	39.53
ELECTRA base NYTK	49.04	20.53	39.76
ELECTRA small wiki	49.02	20.52	39.74
ELECTRA small NYTK	49.04	20.53	39.76

should be exactly one sentence, thus, in Table 5, you can see the F1 scores of the original (orig) and the first generated sentence of the models.

In the case of abstract summarization (see Table 4), the integration of Hungarian models achieved higher performance than the multilingual model. In all cases, the huBERT web gained the best results. As you can see in Table 4, adding index.hu data and applying cleaning methods leads to a performance increase of about 6%.

In the case of H+I, we can see the steps numbers in parentheses that achieved the best results. According to the steps, the huBERT at 450,000 step achieved the best results, much earlier than the other models. In the case of HILBERT, we did not achieve the theoretical optimum, because the rouge values are increasing continuously. As we can see in Table 4, the performance of HILBERT is much lower than the other models. Since the HILBERT is BERT-large, with twice as many parameters as a BERT-base, the model is more robust and the fine-tuning is more difficult. After 47 failed experiments, we could find a set of hyperparameters (see Table 3) that the model could converge with. We believe, that the HILBERT could gain higher results, but we need more experiments to find the best set of hyperparameters to achieve the highest result.

In the case of extractive summarization (see Table 6), all Hungarian models have scored higher than the multi-BERT. As was expected, the Hungarian huBERT web scored the best results. The interesting fact in the results is that our ELECTRA models, which were trained with modest compute, could achieve higher results than huBERT wiki. The ELECTRA models could not outperform huBERT web, just as we expected it would not, after all the huBERT web model was trained on over 9 billion tokens. The result that our ELECTRA models outperformed the huBERT wiki model is significant as the ELECTRA models have much fewer parameters than the BERT base models and can be trained on a single GPU, ideally, within as little as 5 days. It should be noted that training time can be even shorter if the dictionary size is reduced.

We can see some samples in Table 7–10 (see Appendix) which were generated by our abstractive summarization models. Analyzing the samples, we can notice some common features of our models. When the article is long (see Table 8 and Table 9), all of our models extract phrases from the original article, then combine them to generate new sentences. It is similar to extractive models, the difference is that our extractive models choose full sentences from the article and after ranking, give them as results to the user. Generally, the sentences produced by the abstractive models are mostly grammatically correct. All the models generate several sentences, but by the end they "run out" and may leave sentence fragments (see Table 8).

When the article is short (see Table 7 and Table 10), the models show their real abstractive feature, which is to generate passages that the original article did not contain. But in this case, there is too little information in the original article, thus the performance of the output is lower.

Following the samples, we can see the disadvantages of the automatic evaluation metric, such as ROUGE, as well as the problem of using lead as summarization. The ROUGE metric shows only how the generated output is similar to the lead. However, often the function of the lead is to attract attention and not to summarize. In Sample 1. (see Table 7), the article is about damages caused by storms and the payments by the insurers. The lead was only about the insurers it did not even mention the storm and the damages, whereas our models described both topics. This is one of the reasons that in the results (see Table 4) we can see only about 50% recall results.

For more examples visit our demo site[14].

8. Summary

It is concluded that we have created various text summarization tools for the Hungarian language. For building the summarization models, we used different kinds of BERT-based models. For abstractive models, we used the pretrained multilingual cased BERT model as well as the Hungarian monolingual huBERT base and the HILBERT large models.

For extractive summarization, besides the BERT models, we trained our own ELECTRA models. To fine-tune the BERT-based models for summarization tasks, we used the PreSumm tool. The results show that the monolingual Hungarian models outperformed the multilingual model in all cases. The huBERT web model that was trained on 9 billion words could gain the best results both in abstractive and in extractive tasks. Another important result is that our ELECTRA models were trained with less computational demand and they have much less parameters, could gain better results than the huBERT wiki. Another important point of view is that the ELECTRA models are much smaller than the BERT models, which is important for the end users.

This is the first automatic abstractive and extractive text summarization tool for Hungarian that is based on BERT-based neural network technology.

In the future, we would like to experiment with autoregressive methods, such as BART or PEGASUS.

[14] `http://nlpg.itk.ppke.hu/projects/summarize`

References

[1] A. Celikyilmaz, A. Bosselut, X. He, Y. Choi: *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: Association for Computational Linguistics, June 2018, pp. 1662–1675,
doi: https://doi.org/10.18653/v1/N18-1150.

[2] K. Clark, M.-T. Luong, Q. V. Le, C. D. Manning: *ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators*, in: International Conference on Learning Representations, 2020.

[3] Á. Feldmann, R. Hajdu, B. Indig, B. Sass, M. Makrai, I. Mittelholcz, D. Halász, Z. G. Yang, T. Váradi: *HILBERT, magyar nyelvű BERT-large modell tanítása felhő környezetben*, XVII. Magyar Számítógépes Nyelvészeti Konferencia (2021), pp. 29–36.

[4] S. Gehrmann, Y. Deng, A. Rush: *Bottom-Up Abstractive Summarization*, in: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium: Association for Computational Linguistics, 2018, pp. 4098–4109,
doi: https://doi.org/10.18653/v1/D18-1443.

[5] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio: *Generative Adversarial Nets*, in: Advances in Neural Information Processing Systems, ed. by Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, K. Q. Weinberger, vol. 27, Curran Associates, Inc., 2014, pp. 2672–2680.

[6] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, L. Zettlemoyer: *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, Association for Computational Linguistics, 2020, pp. 7871–7880.

[7] C.-Y. Lin: *ROUGE: A Package for Automatic Evaluation of Summaries*, in: Text Summarization Branches Out, Barcelona, Spain: Association for Computational Linguistics, July 2004, pp. 74–81.

[8] Y. Liu, M. Lapata: *Text Summarization with Pretrained Encoders*, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, Hong Kong, China: Association for Computational Linguistics, 2019, pp. 3730–3740.

[9] Y. Liu, I. Titov, M. Lapata: *Single Document Summarization as Tree Induction*, 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: Association for Computational Linguistics, June 2019, pp. 1745–1755,
doi: https://doi.org/10.18653/v1/N19-1173.

[10] M. Miháltz: *OpinHu: online szövegek többnyelv véleményelemzése*, VII. Magyar Számítógépes Nyelvészeti Konferencia (2010), ed. by A. Tanács, V. Varga, V. Vincze, pp. 14–23.

[11] I. Mittelholcz: *emToken: Unicode-képes tokenizáló magyar nyelvre*, XIII. Magyar Számítógépes Nyelvészeti Konferencia (2017), ed. by A. Tanács, V. Varga, V. Vincze, pp. 61–69.

[12] T. Molnár Lengyelné: *Automatic abstract preparation*, 10th International Conference On Information: Information Technology Role in Development (2010), pp. 550–561.

[13] R. Nallapati, F. Zhai, B. Zhou: *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, San Francisco, California, USA: AAAI Press, 2017, pp. 3075–3081.

[14] S. Narayan, S. B. Cohen, M. Lapata: *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: Association for Computational Linguistics, June 2018, pp. 1747–1759,
doi: https://doi.org/10.18653/v1/N18-1158.

[15] D. M. Nemeskey: *Egy emBERT próbáló feladat*, in: XVI. Magyar Számítógépes Nyelvészeti Konferencia, Szeged: Szegedi Tudományegyetem, 2020, pp. 409–418.

[16] D. M. Nemeskey: *Natural Language Processing Methods for Language Modeling*, PhD thesis, Eötvös Loránd University, 2020.

[17] R. Paulus, C. Xiong, R. Socher: *A deep reinforced model for abstractive summarization*, in: Proceedings of the 6th International Conference on Learning Representations, Vancouver, Canada, 2018.

[18] M. Schuster, K. Nakajima: *Japanese and Korean voice search.* In: ICASSP, IEEE, 2012, pp. 5149–5152, isbn: 978-1-4673-0046-9.

[19] A. See, P. J. Liu, C. D. Manning: *Get To The Point: Summarization with Pointer-Generator Networks*, in: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Vancouver, Canada: Association for Computational Linguistics, July 2017, pp. 1073–1083,
doi: https://doi.org/10.18653/v1/P17-1099.

[20] T. Váradi, S. Koeva, M. Yamalov, M. Tadić, B. Sass, B. Nitoň, M. Ogrodniczuk, P. Pęzik, V. Barbu Mititelu, R. Ion, E. Irimia, M. Mitrofan, V. Păiș, D. Tufiș, R. Garabík, S. Krek, A. Repar, M. Rihtar, J. Brank: *The MARCELL Legislative Corpus*, English, in: Proceedings of the 12th Language Resources and Evaluation Conference, Marseille, France: European Language Resources Association, May 2020, pp. 3761–3768, isbn: 979-10-95546-34-4.

[21] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin: *Attention is All you Need*, in: Advances in Neural Information Processing Systems 30, ed. by I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett, Curran Associates, Inc., 2017, pp. 5998–6008.

[22] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, Q. V. Le: *XLNet: Generalized Autoregressive Pretraining for Language Understanding*, in: Advances in Neural Information Processing Systems, ed. by H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, R. Garnett, vol. 32, Curran Associates, Inc., 2019, pp. 5753–5763.

[23] Z. G. Yang, A. Perlaki, L. J. Laki: *Automatikus összefoglaló generálás magyar nyelvre BERT modellel*, XVI. Magyar Számítógépes Nyelvészeti Konferencia (2020), pp. 343–354.

[24] J. Zhang, Y. Zhao, M. Saleh, P. Liu: *PEGASUS: Pre-training with Extracted Gap-sentences forAbstractive Summarization*, in: Thirty-seventh International Conference on Machine Learning, 2020.

[25] X. Zhang, M. Lapata, F. Wei, M. Zhou: *Neural Latent Extractive Document Summarization*, in: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium: Association for Computational Linguistics, 2018, pp. 779–784,
doi: https://doi.org/10.18653/v1/D18-1088.

[26] Q. Zhou, N. Yang, F. Wei, S. Huang, M. Zhou, T. Zhao: *Neural Document Summarization by Jointly Learning to Score and Select Sentences*, in: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Melbourne, Australia: Association for Computational Linguistics, July 2018, pp. 654–663,
doi: https://doi.org/10.18653/v1/P18-1061.

Appendices

Examples of abstractive models

Table 7. 1. example of abstractive models (HVG experiment).

Article Ha a legjobb képes mémek is érdeklik , ide kattintson . Az internetadóról ebben a cikkünkben olvashat bővebben . Ha arra kíváncsi , mit gondol róla a világ , itt megtalálja .
Lead Az internetadóról szóló híreket olvassa el a hvg.hu-n . A magyar internetadóról szóló cikkünkre kattintva olvashatjátok a legfrissebb mémeket , amelyekről itt találjátok a választ . Íme , a legjobb mémek , a mémek és a mémek is . Ha nem akarunk lemaradni a neten , itt találjátok meg a legjobb mémeket . A Az internetadóról szóló mémek mellett az internetadóról is megemlékezett az internetadóról a mémgyár . A mémek mellett a mémek is felkerültek a netadóra . Íme , a legjobbak . Videó az internetadóról . A legjobbak között van az internetadó , a legjobb képes mémek is . a legjobbak között . Videós összefoglaló . hu
huBERT web Az internetadóról szóló mémek mellett az internetadóról is megemlékezett az internetadóról a mémgyár . A mémek mellett a mémek is felkerültek a netadóra . Íme , a legjobbak . Videó az internetadóról . A legjobbak között van az internetadó , a legjobb képes mémek is . a legjobbak között . Videós összefoglaló . hu
huBERT wiki Az internetadóról szóló híreket olvassa el a hvg.hu-n . A magyar internetadóról szóló cikkünkre kattintva olvashatjátok a legfrissebb mémeket , amelyekről itt találjátok a választ . Íme , a legjobb mémek , a mémek és a mémek is . Ha nem akarunk lemaradni a neten , itt találjátok meg a legjobb mémeket . A
multi-BERT Az internetadóról szóló cikkünkben olvashat bővebben az internetadóról . Mutatjuk , mire érdemes figyelni az internetadót . Íme a legújabb adatok . Az internetadóról ebben a cikkben olvashat bővebben . A legjobb képes mémek is érdekesek . És persze nem is gondolnánk , hogy mit gondolnak róla . Írásunkban megmutatjuk , mit gondol a világról . Írja a cikkünket . Teszteljétek magatokat ezzel a kví .

Table 8. 2. example of abstractive models (HVG experiment).

Article Az idei júliusi viharokat követően csaknem hétezer esetet regisztráltak a biztosítók . A két héttel ezelőtti viharokban sok ingatlan sérült meg a fővárosban , Pest megyében , illetve Somogy megyében . Különösen a Balaton déli partja volt érintett , s sokan csak napokkal később fedezték fel , hogy nyaralójuk károsodott . Több társaságnál is arról számoltak be , hogy átcsoportosították kárfelmérőiket a különösen érintett területekre , a szakemberek a kárszemléket elvégezték , s már a vihar másnapján megkezdődtek a kárkifizetések- írja a portfolio.hu . Magyarországon a legtöbb , ingatlanokkal kapcsolatos káresemény a nyári , május- augusztusi hónapokban történik . A biztosítókhoz 2010-2016 között az évnek ebben a periódusában összesen 988 599 kárbejelentés érkezett , amelyekre több mint 73 milliárd forintot fizettek ki a társaságok . A nyári viharok átlagosan mintegy 115 ezer forint összegű kárt tesznek a biztosított ingatlanokban és egyéb javakban , de e téren igen nagy a szórás : sok a kis összegű , úgynevezett „ bagatell-kár ” , de előfordulnak 10 milliós nagyságrendű káresetek is .
Lead 2010 és 2017 között , a május-augusztusi időszakokban csaknem egymillió kárt térítettek a biztosító társaságok . A kifizetett kárösszeg meghaladja a 73 milliárd forintot - tette közzé a Magyar Biztosítók Szövetsége (MABISZ) .
huBERT web Az idei júliusi viharokat követően csaknem hétezer esetet regisztráltak a biztosítók . A viharokban sok ingatlan sérült meg , s sokan csak napokkal később fedezték fel , hogy nyaralójuk károsodott . Több biztosítónál is arról számoltak be , hogy átcsoportosították kárfelmérőiket a különösen érintett területekre , s már a vihar másnapján megkezdődtek a kárkifizetések - írja a portfolio.hu . A viharok átlagosan mintegy 115 ezer forint összegű kárt tesznek a biztosított ingatlanokban és egyéb javakban , de előfordulnak 10 milliós nagyságrendű káresetek is . Magyarországon a legtöbb , ingatlanokkal kapcsolatos káresemény a nyári , május- augusztusi időszakban .
huBERT wiki Az idei júliusi viharokat követően csaknem hétezer esetet regisztráltak a biztosítók . A Balaton déli partjának volt érintett , s sokan csak napokkal később fedezték fel , hogy nyaralójuk károsodott . Több társaságnál is arról számoltak be , hogy átcsoportosították kárfelmérőiket a különösen érintett területekre , a szakemberek a kárszemléket elvégezték , s már a vihar másnapján megkezdődtek a kárkifizetések . Magyarországon a legtöbb , ingatlanokkal kapcsolatos káresemény a nyári , május-augusztusi hónapokban történik .
multi-BERT Az idei júliusi viharokban sok ingatlan sérült meg a fővárosban , Pest megyében , illetve Somogy megyében . Különösen a Balaton déli partja volt érintett , s sokan csak napokkal később fedezték fel , hogy nyaralójuk károsodott . Több társaságnál is arról számoltak be , hogy átcsoportosították kárfelmérőiket a különösen érintett területekre , a szakemberek pedig a kárszemléket elvégezték , s már a vihar másnapján megkezdődtek a kárkifizetések .

Table 9. 3. example of abstractive models (H+I experiment).

Article Kedd délután a rendőrség megerősítette az Index nek , hogy reggel elfogtak a rendőrök egy férfit Budapesten , aki lőfegyverrel a kezében álldogált a Nagyvárad téren a 24-es villamos megállójában . A lapnak több szemtanú is beszámolt az esetről , a rendőrség pedig azt közölte , hogy kedd reggel 7 óra 12 perckor bejelentést kaptak arról , hogy a Nagyvárad tér 1. szám alatti villamosmegállóban egy fegyver van egy ember kezében . Lövés nem dördült , és senki nem sérült meg , a járőrök perceken belül a helyszínre érkeztek , és intézkedés alá is vontak egy 49 éves budapesti férfit . P. Lászlónál egy gáz-riasztó fegyvert találtak , ezért garázdaság és lőfegyverrel kapcsolatos szabálysértés miatt előállították , és eljárást kezdeményeztek vele szemben .
Lead Lövés nem dördült , a férfit elfogták .
huBERT web Garázdaság és lőfegyverrel kapcsolatos szabálysértés miatt előállították P. Lászlót , akinél gáz-riasztó fegyvert találtak , ezért garázdaság és lőfegyverrel kapcsolatban szabálysértés miatt eljárást kezdeményeztek vele szemben . Lövés nem dördült , senki nem sérült meg , de a rendőrök perceken belül elfogták a férfit . P. Lászlónál egy gáz-riasztó fegyvert is találtak . Garázdaság miatt előállították , és eljárást kezdeményeztek ellene . Garázdaság és lő fegyverrel kapcsolatos szabálysértése miatt előállították . A férfi ellen eljárást kezdeményeztek , és eljárás indult ellene . Videó
HILBERT A férfi fegyverrel a kezében sétált a 24-es villamosmegállóban , a rendőrök perceken belül elfogták , és kezdeményezik előzetes letartóztatásukat . A rendőrség eljárást indított vele szemben , és eljárást indítottak vele szemben . A férfi ellen eljárás indult . A rendőrség szerint egy gáz-riasztó fegyvert is találtak a rendőrök , és elrendelték az előzetes letartóztatását . Az MTI úgy tudja , hogy a férfi egy gáz-riasztó fegyverrel is felszerelkezett a 24-es villamos megállójában . Videó is van a helyszínről , a rendőrség a hvg.hu-nak arról számolt be , hogy kedd reggel fogtak el egy férfit , aki lőszerrel a rendőrök elől menekülve próbált meg egy embert a 24-es villamoson . hu hu
multi-BERT Kedd reggel elfogták a rendőrök azt a férfit , aki lőfegyverrel a kezében álldogált a Nagyvárad téren a 24-es villamos megállójában . Lövés nem dördült , és senki nem sérült meg , a járőrök perceken belül a helyszínre érkeztek , és intézkedés alá is vontak egy 49 éves budapesti férfit . P. Lászlónál egy gáz-riasztó fegyvert találtak , ezért garázdaság és lőfegyverrel kapcsolatos szabálysértés

Table 10. 4. example of abstractive models (H+I experiment).

Article A Daily Mail Twitter-oldalán közzétett videó szerint most épp nem történt nagy baj , de ez könnyen történhetett volna másképp is .
Lead Valószínűleg nem gondolták át igazán ezek a fiatalok mit tesznek .
huBERT web A Daily Mail Twitter-oldalára került fel egy videó arról , ahogy az egyik brit autópályán egy motorcsónaknaknak csapódott . A videón jól látható , hogy a sofőrnek nem sok esélye van a balesetre , de a videón jól látszik , hogy nem is kell aggódnia . És nem is ez volt az első ilyen eset . A videóban rögzítették , ahogy a britek egy hatalmasat ugranak a fejük előtt . Az alábbi videón látható , hogyan történt a dolog , és hogyan lett volna ez a videóból . A videón látható .
HILBERT A Magyar Kétfarkú Kutya Párt szerint most már nem is történt nagy baj , de ez így is elég lett volna . nem ez volt az első eset , hogy nem lett komoly baj . „ Ez történt volna ” - reagált a párt . Videón az is látszik , hogy a párt szerint nem történt baj . Igaz , ez még így is így lett volna , ha nem is lett volna baj , az eredmény nem lett volna ennyire egyszerű .
multi-BERT A Daily Mail videója szerint nem történt nagy baj . A videó könnyen előfordulhatott volna , hogy egy hatalmas bajba került . Nem kellett volna . Videó . Szerencsére nem történt baj , de ez könnyen történhetett volna másképp is . De nem . Videón . Mutatjuk . Szerencsésen volt . Videót . És . Munkatársunk . Sőt . Mutatja !Videó !Szavazzon !! A hvg.hu hu . A