Acta Linguistica Academica 69 (2022) 4, 474–500
DOI: 10.1556/2062.2022.00577

AKADÉMIAI KIADÓ

Neural text summarization for Hungarian

ZIJIAN GYŐZŐ YANG[1,2*]

[1] Hungarian Research Centre for Linguistics, Hungary

[2] MTA-PPKE Hungarian Language Technology Research Group, Hungary

Received: April 21, 2022 • Accepted: October 1, 2022
Published online: November 29, 2022

ABSTRACT

One of the most important NLP tasks for the industry today is to produce an extract from longer text documents. This task is one of the hottest topics for the researchers and they have created some solutions for English. There are two types of the text summarization called extractive and abstractive. The goal of the first task is to find the relevant sentences from the text, while the second one should generate the extraction based on the original text. In this research I have built the first solutions for Hungarian text summarization systems both for extractive and abstractive subtasks. Different kinds of neural transformer-based methods were used and evaluated. I present in this publication the first Hungarian abstractive summarization tool based on mBART and mT5 models, which gained state-of-the-art results.

KEYWORDS

extractive summarization, abstractive summarization

1. INTRODUCTION

Processing large amounts of textual data in our everyday life with manual tools is becoming increasingly difficult given the scale of the data to be analysed. For instance, any company, public institution or international organization accumulates an enormous amount of text data in the course of their operations, which has to be stored, curated, and processed according to the predefined requirements of the stakeholders. Often it is important to be able to get a quick insight into the content of a document with hundreds of pages in order to find relevant

[*] Corresponding author. E-mail: yang.zijian.gyozo@nytud.hu, yang.zijian.gyozo@itk.ppke.hu

information or to support executive decision-making without reading through the entire document. Thereby, a key challenge is to extract the essence of the data from the huge body of texts. The application of automatic methods for extracting or summarizing can lead to significant optimization of running time and costs, which results in an increasing demand for automatic information extraction applications. Automated text summarization is a particularly pressing, yet-to-be-solved issue in the case of Hungarian language. Automated text summarization is the process of text compression in a document using a system that can process information by prioritizing and retaining information that is especially essential to have a short overview of the content. Technologies that generate summaries take into account variables such as length, style, or syntax. Text summarization from a human perspective is like taking a bit 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 text while capturing the meaning of the original article. Abstractive methods are usually more complex because the machine-based solution has to analyze the text first, then highlight the most important information from it, then learn the relevant concepts and finally, it has to construct cohesive summaries. Extractive text summarization does not generate any new text, it only uses words already present 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 the order of importance, and selecting the most important components of a document to create a summary.

In this paper, both extractive and abstractive research are introduced and own experiments are described for Hungarian. Furthermore, I present in this publication the first Hungarian abstractive summarization tool based on mBART and mT5 models, which gained state-of-the-art results.

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 (Nallapati, Zhai & Zhou 2017), which uses an RNN encoder to solve the problem. Another method called Refresh (Narayan, Cohen & Lapata 2018) is based on the Rouge metric, which is used to rank sentences in the text using the reinforcement learning method. The goal of Latent (Zhang et al. 2018) 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 (Liu, Titov & Lapata 2019) 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 (Zhou et al. 2018) approaches the problem by scoring and selecting sentences from the original text.

A combination of extractive and abstractive summary generation methods revealed that higher ROUGE (Lin 2004) scores could be achieved if an extractive summary of the original text was included in the training pipeline of the abstractive summary generation process.

The neural network-based abstractive summarization approaches the problem as a transformation task, during which a sequence is converted to 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 (See et al. 2017) 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 (Celikyilmaz et al. 2018) is an agent-based approach, in which the task of encoding a long text is shared among multiple collaborating agents, each of them responsible for 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 (Paulus, Xiong & Socher 2018) 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 (Gehrmann, Deng & Rush 2018) 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 (Liu & Lapata 2019) model was considered as the state-of-the-art tool in 2019. It requires a pre-trained BERT model to train extractive and abstractive summarization models. Pre-training a BERT model requires huge data and computational capacity. Optimally, we can choose the PreSumm model, since recently a number of BERT models have been created for Hungarian language. We can use the multilingual BERT,[1] which includes Hungarian among the covered languages. In addition to that, there are two Hungarian monolingual BERT base models built by Nemeskey (Nemeskey 2020b) that we could use for our research purposes.

So far autoregressive methods achieved the best results in the field of summarization. Autoregressive models rely on the decoder of the transformer model and apply an attention mask on the top of the full sentence so the model can only look at the tokens before the full availability of the text being investigated. This method achieved higher results on many text generation tasks (Yang et al. 2019). The BART model (Lewis et al. 2020) is a denoising autoencoder for pre-training sequence-to-sequence models, which is trained to corrupt text with a built-in arbitrary noising function followed by text reconstruction. This model is especially effective to fine-tune summarization tasks.

Currently, the state-of-the-art tool for summarization is the PEGASUS (Zhang et al. 2020) system. In the procedural pipeline of PEGASUS, important sentences are first removed/masked from an input document and subsequently concatenated together as one output sequence from the remaining sentences, similarly to the concept of creating an extractive summary.

Constructing highly accurate summaries requires models that are able to incorporate the richness of a given language, moreover, these models need to understand the underlying meaning in the original texts. A collection of research examples for various languages has already been established in this direction. For example, the mBART model was adapted to

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

summarize texts from Russian news. The experimentation with the mBART model optimized for Russian language revealed that it was possible to achieve exceptionally good results in text summarization, despite that mBART was originally not designed for Russian. mBART reached outstanding results according to automated evaluation using ROUGE, BLEU, and METEOR metrics. This was further supported by the human annotation of the generated summaries compared with human references. In the course of the previously mentioned comparison of machine-generated versus human summaries mBART turned out to be a winner in more than 73 percent of the cases in terms of extractiveness. Another example for Russian language has been developed in the area of headline generation. A Transformer model with 6 layers and 8 heads was trained to construct summaries of Russian news articles and subsequently, to generate headlines. The trained model outperformed other models and set new state-of-the-art standard for ROUGE-2 benchmark (Malykh, Chernyavskiy & Valyukov 2020). A critical aspect of any text summarization method is to have text summarization datasets. The availability of text summarization datasets for other languages than English is still scarce, therefore Gusev was motivated to improve this situation by expanding the space of Russian text summarization datasets (Gusev 2020).

A neural network model using was constructed from a BERT encoder and a transformer-based decoder to summarize Japanese texts. The established model performed similarly to the pointer-generator network included in the study for comparison. The authors observed two key problem areas that occurred during the model testing: these are word repetition and unknown vocabulary. To mitigate the effects of the issues on summary text quality, the repeat block and WordPiece were applied. Consequently, the repeat block regenerated the summaries without the repeated words and WordPiece divided the unknown words into sub-words (Iwasaki et al. 2020).

For many low-resource languages, the lack of appropriate datasets is still a major obstacle. Hasan et al. published a text summarization dataset called XL-Sum that contains 1 million summary-article pairs in 44 languages using the BBC website as a source. Furthermore, the research group created an automated crawling tool that extracts article-summary pairs from BBC making it possible to expand the dataset over time. Finally, multilingual summarization has been performed for various languages. Both human and automatic evaluation confirmed that the generated summaries are highly abstractive and successfully capture the meaning and key ideas in the original article. Taken together, XL-Sum can be considered as an important milestone in the history of abstractive summary generation opening up new ways for low-resource languages (Hasan et al. 2021).

Abstractive summarization represents a key challenge in the case of moderately resourced languages, such as Hungarian. For Hungarian language, a cumulative effort resulted in the development of an abstractive summarizer. For Hungarian, the OpinHu system has a summary function (Miháltz 2010). The system uses keywords and text context to extract information. Lengyelné Molnár Tünde (Lengyelné Molnár 2010) examined the possibilities and limitations of the automatic generation of research abstracts. Using the PreSumm (Liu & Lapata 2019) tool, Yang, Perlaki & Laki (2020) created the first extractive summarization tool for Hungarian. Also based on Presumm tool, Yang et al. (2021) created the first abstractive summarization tool as well. Yang (Yang 2022c, b) in his research, different experiments were done to pre-train and fine-tune Hungarian monolingual BART and GPT-2 models to summarization task.

Besides the research of Yang et al., Makrai et al. (2022) did experiments with huBERT-based neural abstractive summarization. A huBERT model with an encoder-decoder architecture was fine-tuned on the ELTE.DH corpus, which is a collection of articles from former Hungarian news portals. Additionally, Automated Speech Recognition (ASR) was used to generate transcripts from audio files, which made it possible to compare the different summarization outcomes between written and spoken sources. It is important to highlight that ASR can introduce errors that are transmitted further down the analysis pipeline. In order to evaluate the performance of the model, the ROUGE metrics were applied. In comparison to manually transcribed and punctuated texts, the ASR-derived summaries received similar or even higher ROUGE scores. Interestingly, the summarization of weather forecasts resulted in the highest ROUGE-1 and ROUGE-L F1 scores, which suggests that a highly optimal summary length could be achieved.

As a result of my pioneering work, I present in this publication the first Hungarian abstractive summarization tool based on mBART and mT5 models.

3. PRE-TRAINED NEURAL LANGUAGE MODELS FOR TEXT SUMMARIZATION

3.1. BERT, mBERT

BERT (Bidirectional Encoder Representations from Transformer) is an attention-based, multilayer, bidirectional Transformer encoder (Vaswani et al. 2017). The BERT model was trained on masking and next sentence prediction tasks. During the masking procedure, 15 percent of the words in the text are randomly masked, then the system is trained to acquire the correct word. In the course of the next sentence prediction task, the model is provided with two sentences, then the model creates a judgment, whether the two accepted sentences are next to each other in the original text or just two randomly selected sentences. We applied WordPiece tokenizer (Schuster & Nakajima 2012) to limit the size of the dictionary and to solve the out-of-vocabulary words problem. Next in line is the fine-tuning process, when the pre-trained BERT model is further trained on a specific downstream task with a feed-forward network.

BERT models have not been exclusively trained on English, but on several other languages as well, which is a key advantage in comparison to other models. The AI Research Group at Google contributed to the groundbreaking efforts to train two multilingual models:[2] a lowercase and a normal one. The first 104 languages with the largest Wikipedia representation were selected to be included in the training procedure. The size of Wikipedia content substantially varies between languages, e.g. the English Wikipedia itself accounts for nearly 20 percent of the data, thereby sampling was controlled by normalization to overcome this issue. Next, all languages were tokenized in an identical manner to English, which involved four steps: lowercasing, accent removal, punctuation, and whitespace handling. Training the normal model also followed these steps without the lowercasing part. WordPiece tokenization and dictionary is able to handle cased and unknown words as well. Importantly, Hungarian language was represented amongst the 104 languages covered.

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

BERT models have great potential to improve the efficiency of summarization tasks. For example, Abdel-Salam and Rafea experimented with a BERT model, in which the attention layers were replaced with grouped convolutional layers and tested its performance in text summarization in comparison with the baseline BERT model. Interestingly, the authors found that the modified BERT model retained 98 percent of the baseline model performance. This finding is especially intriguing in light of the fact that the modified model operated with only half as many parameters as the baseline BERT (Abdel-Salam & Rafea 2022).

3.2. ELECTRA, ELECTRIC

ELECTRA (Clark et al. 2020a) is based on the concept of Generative adversarial network (GAN) (Goodfellow et al. 2014) method. The GAN includes two main components: a generator and a discriminator. During training, the generator randomly generates vector representations, which give rise to the output. Next, the real output data is presented to the system with the aim to improve the performance of random vector generation. Thereby, the generator gradually becomes "smarter" and generates an output that closely matches the real output by the end of the training session. The discriminator is trained to judge whether a particular word is the original one or a replacement. In the course of this process, the discriminator receives data from the real corpus, and simultaneously obtains the output data from the generator. By the end of the training. the generator can generate content that is similar to the original/real content and the discriminator acquires capabilities to distinguish between a fake/erroneous content and a real/correct content.

ELECTRA is a modified GAN method for training language models. The difference compared to the operating concepts of BERT model (and the original GAN) lies in that ELECTRA does not predict the original word behind the masked word, instead, the generator randomly generates words for the masked words and then the discriminator has to make a judgement if the words given by the generator are the original words or randomly generated words. Thus, the generator learns step-by-step what actual words match to the place of the masked words, while the discriminator is trained to distinguish between the presented inputs, whether the actual text was built with real or fake words. After the training is completed, the generator is discarded and only the discriminator is kept for subsequent fine-tuning.

Experimental work revealed that ELECTRA underachieved compared to BERT and RoBERTa models in extractive summary generation of texts from the arxiv dataset demonstrated by the lower ROUGE scores given to the summaries generated by ELECTRA (Tretyak & Stepanov 2020).

Electric was developed as an attainment of the cloze task using an energy-based model (Clark et al. 2020c). Electric shares a high level of similarity with ELECTRA models. In contrast to BERT models, Electric does not apply masking or normalization based on softmax concepts. Instead of the application of the above-mentioned principles, Electric simply assigns a so called energy score to each token position and the distribution of possible tokens is calculated in a subsequent step. Furthermore, the training of the Electric involves a Noise-Contrastive Estimation (abbreviated as NCE) (Gutmann & Hyvärinen 2010). There are two key differences between ELECTRA and Electric models. Firstly, the ELECTRA uses a masking algorithm, while Electric applies a two-tower cloze model to approach the noise distribution issue (Baevski et al. 2019) The two-tower cloze model manages noise distribution by applying the context to both sides of the tokens, which requires two transformers, one operating left-to-right and the other in the opposite direction. Secondly, ELECTRA determines likelihood scores only for the masked tokens, while in the case of Electric these scores are computed for all input tokens at the same time. Taken together, Electric encompasses several updated features compared to its predecessor, however, a major disadvantage of Electric over ELECTRA is the rigidity of Electric in the choice of noise distribution, which might exclude it from applications that necessitate flexibility in that sense (Clark et al. 2020b).

3.3. BART, mBART

BART (Lewis et al. 2020) is a Transformer-based denoising autoencoder with an applicability to pre-train sequence-to-sequence models. BART has an encoder-decoder architecture that is capable of reconstructing the original text by predicting the corrupted parts using a 'noised' source text as input. BART's setup is similar to BERT (Devlin et al. 2019), however, with major differences in its architecture. Notably, one such difference is the presence of an additional cross-attention of the layers over the final hidden layer in BART, which is not present in BERT. Moreover, BERT has an additional feed-forward network prior to the word-prediction step, which cannot be found in BART. BART offers a wide range of applicable noising schemes providing compatibility with basically any type of document corruption, which is usually not the case with other denoising autoencoders. The BART can be considered as a hybrid of BERT and GPT models. The BART differs from the BERT in the following denoising tasks:

- Token Masking: random tokens are sampled and replaced with [MASK] elements.
- Token Deletion: random tokens are deleted from the input.
- Text Infilling: a number of text spans (Joshi et al. 2020) are sampled, with span lengths drawn from a Poisson distribution. Each span is replaced with a single [MASK] token.
- Sentence Permutation: a document is divided into sentences based on full stops, and these sentences are shuffled in a random order.
- Document Rotation: a token is chosen uniformly at random, and the document is rotated so that it begins with that token.

Additionally, BART can be efficiently fine-tuned to be optimal for various downstream tasks, such as sequence and token classification, sequence generation, and machine translation. The performance of BART can be especially high in large-scale pre-training, for instance, in discriminative tasks like SQuAD (Rajpurkar et al. 2016) and GLUE (Wang et al. 2018). In these benchmarks BART performance is comparable to that of RoBERTa. Importantly, BART outperforms all previously established models in summarization tasks. Accumulating evidence suggests that BART performs the best when applied for Natural Language Generation (NLG), but achieves remarkable results in translation and comprehension tasks as well. Abundant evidence suggests that BART achieves its best, when applied for the Natural Language Generation (NLG), but it also performs remarkably well in translation and comprehension tasks. Additionally, there have been trials to implement BART as a tool to summarize texts. In a recent example, the content of COVID-related research papers was summarized using BART.[3]

[3] https://deeplearninganalytics.org/summarization-of-covid-research-papers-using-bart-model

Furthermore, a two-stage text summarization approach was applied to summarize CNN/DailyMail articles, in which the first round was a BERT-based extractive summary generation that was fed to the abstractive part with BART and GPT-2 models. The two-stage system outperformed Lead-3 and BERTSumEXT models.[4] Experimental work also revealed that hierarchical BART (Hie-BART) could outperform the non-hierarchical BART model trained on the CNN/DailyMail dataset, demonstrated by an increase of 0.23 points in the F-score of ROUGE-L Akiyama, Tamura & Ninomiya (2021).

In the course of our current research, we did experiments with BART base models. Two different BART models were trained for English by the Meta Research:

- BART base: 12 layers; hidden layer size: 768; 139M parameters.
- BART large: 24 layers; hidden layer size: 1,024; 406M parameters.

The mBART (Liu et al. 2020) is a denoising autoencoder model pre-trained on multiple languages and it is based on the seq2seq concept. It is usually applied to improve the performance of both supervised and unsupervised machine learning. The mBART follows the BART scheme in its architecture. The authors of the model put an emphasis on multilinguality during pre-training, then the model was fine-tuned in a bilingual setting. As for the pre-training, the CC25 (Wenzek et al. 2020; Conneau et al. 2020) corpus was used, which contains 25 languages and the texts are extracted from the CommonCrawl database. The multilingually pre-trained model was used for both sentence- and document-level machine translation. It is particularly important to point out that only with the application of the seq2seq concept could improve the quality of document-level machine translation, this is a significant step forward compared to previous research work (Miculicich et al. 2018; Li, Jiang & Liu 2019). The experimental results with the mBART model highlight the true potential of multilingual pre-training with an applicability potential for transfer learning.

The mBART model has an extended version, which is called mBART-50 (Tang et al. 2020). Using the original mBART model extra 25 languages were added to support multilingual machine translation models of 50 languages. mBART models, even the mBART-50 model, do not include Hungarian language capabilities, but taken the advantage of sentence piece tokenization (?), we could adapt this model for Hungarian.

3.4. GPT models

Models based on the decoder-only architecture, such as the generative model pre-training (abbreviated as GPT) have substantially accelerated the development of Natural Language Processing. Recently emerged semi-supervised learning algorithms opened up new opportunities to approach the current problems of text processing, as demonstrated by a pioneer project led by the OpenAI research group. Specifically, applications like sequence labeling and text classification gained significant interest from both the academic and the industrial sectors. Furthermore, accumulating theoretical and practical evidence suggests that text embedding methods can significantly improve the performance of language models. The analysis of texts can be

[4] https://pythonawesome.com/two-stage-text-summarization-with-bert-and-bart/

performed at various levels. For example, the OpenAI[5] research group focused on a level of semantic investigation that is superior to the level of words, which facilitates vector representation of the higher-order units. Additionally, the application of unsupervised pre-training can foster the capturing of even more sophisticated linguistic information, which can be extrapolated to long-term information extraction by choosing the right Transformer. The application of auxiliary training objectives can increase performance as it is highlighted in the work of Rei et al. (Rei 2017).

Unsupervised language models can solve language processing tasks if the training is performed on sufficiently large datasets. In order to illustrate the scale of the required data volume, WebText can be given as a good example with its content acquired from millions of websites. Former studies proved the importance of zero-shot learning strategies and their crucial role in language processing tasks. Importantly, GPT-2 (Radford et al. 2019), which is a transformer model comprising 1.5 billion parameters achieved stet-of-the-art performance in 7 out of 8 tasks in a zero-shot setting. GPT-2 experimentation raises pivotal questions about how training models on large datasets can improve language model performance, thereby numerous studies have been made to deliver solid experimental data to gain a more informed insight into the underlying processes. For example, Jozefowicz et al. experimented with RNN (Recurrent Neural Network)-based language models on the 1 Billion World Benchmark. Interestingly, one of their models reached an outstanding improvement in perplexity (from 51.3 to 30.0), and an ensemble of their models set a new state-of-the-art result in perplexity reduction (from 41.0 to 23.7) (Jozefowicz et al. 2016). Bajgar et al. applied an Attention-Sum Reader model trained on the Book Test dataset (similarly structured to the Children's Book Test (CBT) dataset, but approximately 60 times bigger). The model trained on the Book Test reached more accurate results on the Children's Book Test suggesting that training the same model on a larger dataset will lead to superior performance compared to the setup when it is only trained on a smaller dataset (Bajgar, Kadlec & Kleindienst 2017). Hestness et al. hypothesized whether model performance could be improved with increased model capacity and datasets of bigger size. The findings largely correlate with the trends observed in the case of GPT-2 experimentation (Karpathy, Johnson & Fei-Fei 2016). Finally, Liu et al. trained the model with decoder-only architecture to generate Wikipedia articles and surprisingly it also acquired the capability to translate names between different languages (Liu et al. 2018).

There are multiple strategies described in the literature that can enhance the performance of language models in solving generative or other language processing tasks: (1) increasing the number of parameters and computational power simultaneously, (2) inclusion of more parameters, or (3) the augmentation of computational capacity.

The GPT-3 model (Brown et al. 2020) falls into the first strategy: it improves model performance by the increase of parameters and computational capacity at the same time. The model has 175 billion parameters and it achieves state-of-the-art results in several tasks without any fine-tuning procedure. In order to train language model in an unsupervised manner, it is crucial to have large datasets that are not annotated. For example, Wikipedia, Gigaword (Graff et al. 2003), the non-public Google News corpus, the RealNews database (Zellers et al. 2019) or the WikiText (Merity et al. 2017) are well-known examples of such

[5] https://openai.com

large-scale datasets in the literature. Internet scraping is getting more and more common to generate large datasets. One of the most prominent representative dataset in this category is the Common Crawl. However, despite the fact that web scraping can give rise to large datasets, the filtering and the cleaning of data can be still challenging and often claimed as a limiting factor. The combination of data from multiple sources to create training datasets is getting widespread as well.

Numerous examples exist that try to enhance the performance of transformer-based models by increasing the number of parameters. In-depth experimental work was dedicated to research the relationship between the number of parameters and performance including models with 300 million (Devlin et al. 2019), 8.3 billion (Shoeybi et al. 2020) and 11 billion (Raffel et al. 2020a) parameters. In an attempt to improve model performance exclusively with parameter number alterations i.e. information storage capacity augmentation without manipulating of computational capacity. A highly relevant example of such efforts is by Aharoni et al., who trained a model with 50 billion or 100 million parameters and monitored how translation of TED talk transcripts is affected by these manipulations (Aharoni, Johnson & Firat 2019). Alternatively, adaptive communication time (ACT) can serve as an opportunity to increase the efficiency of RNN-based models in a way that the number of computational steps is optimized between the input and the output (Graves 2017). The Universal Transformers that are a hybrid solution between RNN-based and Transformer models, similarly increase computational power (Dehghani et al. 2019).

Accumulating evidence suggests that the advanced machine learning capabilities represented by GPT models could be efficiently exploited in the field of text summarization. Numerous examples demonstrate that powerful performance could be achieved, when GPT was applied to summarize medical conversations Chintagunta et al. (2021), judicial case summarization in Chinese Liu, Wu & Luo (2021) and summarization of hotel reviews Basols (2021).

GPT models can be characterized in general by the usage of BPE coding as dictionary. The current 3+1 GPT models in comparison:

- GPT: 12 layers, 12 attention heads; 768 word embedding size; 512 text length; 117 million parameters;
- GPT-2: 48 layers, 12 attention heads; 1,600 word embedding size; 1,024 text length; 1.5 billion parameters;
- GPT-3: 96 layers, 96 attention heads; 12,888 word embedding size; 2,048 text length; 175 billion parameters;
- GPT Neo (Brown et al. 2021): mesh-tensorflow library implementation in order to train GPT-3 type models

We carried out experiments with a GPT-2 model in the course of our research.

3.5. T5, mT5

The T5 (Text-To-Text Transfer Transformer) (Raffel et al. 2020b) is a model and framework developed by the Google research team and offers new ways to approach central issues in natural language processing. The core concept behind T5 is transfer learning, in the course of which the language model is trained on a data-rich task followed by a fine-tuning step for a specific task.

Specifically, general knowledge is acquired during the pre-training phase that is transferred and further applied to the specific tasks. The T5 project applies transfer learning principles in the contextual setup of the seq2seq approach. The original idea was that a whole range of language processing tasks (translation, question answering, and classification) should be seen as a text-to-text problem, therefore both the input and the output are in text format. The great advantage of the text-to-text paradigm is that its applicability is significantly extended, since it can be used for practically any NLP task, from machine translation and summary generation to question answering, or even sentiment analysis.

The application of specifically suited corpora is required to support the demands of transfer learning concepts. For this reason, the Colossal Clean Crawled Corpus (abbreviated as C4) was generated, which is a filtered collection of hundreds of gigabytes of the World Wide Web in English. The C4 corpus is based on the Common Crawl 5 database. Another important characteristic of transfer learning methods is that for the pre-training, non-labelled datasets are required. In addition to the previously detailed requirements, the applicable corpora have to be large enough, diverse, and high-quality. To illustrate this, the C4 corpus is two times the size of Wikipedia resulting in significantly more data provided for pre-training. Five different models have been created as a part of the T5 framework, which differ in the number of parameters: Small (300 million parameters), Base (580 million parameters), Large (1.2 billion parameters), XL (3.7 billion parameters), XXL (13 billion parameters).

Taken together, a highly efficient framework has been created as a result of the T5 project as it is demonstrated by the outstanding results achieved by the T5 models. One of the models with 11 billion parameters set new state-of-the-art at several benchmarks, for example in GLUE, SuperGLUE, SQuAD, and CNN/Daily Mail reference tasks. The mT5 (Xue et al. 2021) extends the English-based T5 to several other languages. The authors tried to preserve the structural characteristics of T5, thereby mT5 inherited the text-to-text problem approach and the general pre-training on large datasets.

In order to train mT5, the mC4 corpus was used, which is the multilingual version of C4 with texts from 101 different languages, including Hungarian amongst them. A common problem with multilingual models is the under-representation of certain languages and the over-representation of others. If a certain language is underrepresented in the corpus, inappropriate fitting will occur due to the higher sampling rate. In order to overcome this issue, the authors applied a frequency-based sampling procedure similarly to previous research efforts (Devlin et al. 2019; Aharoni, Johnson & Firat 2019). Furthermore, it was therefore necessary to apply a larger dictionary consisting of 250,000 word pieces, since the mT5 model was trained on a corpus consisting of more than 100 languages.

6 tasks from the XTREME multilingual reference system (Hu et al. 2020) were applied to assess the performance of the mT5 with tasks like sentence pairing, noun recognition, and question answering. For example, mT5-XXL (the largest model in the mT5 framework) achieved the highest performance in question answering. The mT5 can be successfully applied in a multilingual context and the models can achieve outstanding results in a number of reference tasks and the applicability of the framework can be extended to other areas as well. Pre-trained mT5 has been applied to successfully generate abstractive text summaries in a range of languages, including Persian (Farahani, Gharachorloo & Manthouri 2020), Arabic Fuad & Al-Yahya (2022), and Stankevičius & Lukoševičius (2021). Since mT5 includes Hungarian language as well, we could apply the mT5 small model for summary generation in our research.

4. PRE-TRAINED NEURAL LANGUAGE MODELS FOR HUNGARIAN

The first Hungarian BERT model was created and published by Nemeskey (Nemeskey 2020b), which is called huBERT.[6] Three huBERT models were trained:

– huBERT: BERT base model trained on Hungarian Webcorpus 2.0[7]
– huBERT Wikipedia cased: cased BERT base model trained on Hungarian Wikipedia
– huBERT Wikipedia lowercased: lowercased BERT base model trained on Hungarian Wikipedia

The huBERT models are highly efficient, which is highlighted by the fact that these models continue to maintain their position in achieving state-of-the-art results in name entity recognition and noun phrase chunking tasks (Nemeskey 2020a).

The first BERT large model (HILBERT) has been built by Feldmann et al. (2021) for Hungarian. The HILBERT model could not outperform huBERT in downstream tasks.

Earlier this year, the first English-Hungarian bilingual GPT-3 generative language model, the HILANCO-GPTX,[8] has been trained by the HILANCO consortium.

In recent years, further experimental neural language models have been successfully trained for Hungarian: HIL-ELECTRA (Yang & Váradi 2021), HIL-ELECTRIC (Yang & Váradi 2021), HIL-RoBERTa (Yang & Váradi 2021), HIL-ALBERT,[9] HILBART (Yang & Váradi 2021), NYTK-BART models (Yang 2022c), NYTK-GPT-2 (Yang 2022b).

The HIL models (except HILBERT) are usually experimental small models with a limited amount of training data. They could not outperform the huBERT, but they could achieve comparable results.

The first Hungarian autoregressive neural language models (NYTK-BART and NYTK-GPT-2) were trained by Zijian Győző Yang, different BART models and a GPT-2 model were trained. Due to the fact that these models are also experimental models, they could only achieve the best results in the generative tasks. Using these models the first neural-based Hungarian poem and news generator models were trained.

5. CORPORA AND EVALUATION METRICS

In our experiments we used five different corpora for the summarization tasks: HVG, index.hu, HI (HVG + index.hu), MARCELL, NOL. In Table 1, you can see the main characteristics of the corpora. HVG[10] and index.hu[11] are active online daily news sites, NOL[12] is a closed (in 2016) daily news site. MARCELL Váradi et al. (2020) is a corpus that contains legal documents.

[6] https://hlt.bme.hu/en/resources/Hubert

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

[8] https://hilanco.github.io/home.html

[9] https://hilanco.github.io/models/albert.html

[10] https://hvg.hu

[11] https://index.hu

[12] http://nol.hu

Table 1. Main characteristics of the corpora

Year	HVG 2012–2020	index.hu 1999–2020	HI –	MARCELL 1991–2019	NOL 1999–2016
Documents	480,660	183,942	559,162	24,747	397,343
Token	129,833.741	104,640.902	159,131,373	28,112,090	168,789,330
Type	5,133,030	3,921,893	3,053,703	450,115	2,589,211
Avg token # – article	246.27	496.27	265.17	1124.82	384.52
Avg token # – lead	12.43	22.33	29.97	11.22	39.71
Avg sentence # – article	23.74	35.76	11.40	49.26	17.36
Avg sentence # – lead	1.46	2.23	1.57	1.00	1.86

In the case of HVG, index.hu and NOL, the body of the articles taken from the daily online newspaper, as well as the corresponding leads, representing the summaries (a sample can be seen in Table 2). Two corpora were built from HVG and index.hu. In the first version, only the HVG documents were used. In the second version (HI corpus), the HVG and the index.hu articles were merged. In the case of MARCELL, the legal documents were used as source and each of these has one short sentence topic description that we used for the target summary.

In the experiment with HVG, NOL and MARCELL corpora, there were no pre-processing steps. In the case of HI corpus, we applied cleaning processes. The cleaning and normalizing aspects for HI are as follows:

- 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
- removed irrelevant articles or text parts: e.g.: "Follow us on Facebook", "Edited: [NAME]", "Click for more details", "Start a Quiz", etc.

As for the evaluation of extractive and abstractive summarization models, the ROUGE (Lin 2004) metrics were applied. 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 an unigram, while ROUGE-2 is a bigram coverage calculation algorithm. ROUGE-L examines the longest common word sequence at the level of paragraphs and sentences. We used the ROUGE-1, ROUGE-2, and ROUGE-L metrics in our evaluation. The result tables have been compiled using the following format:

- ROUGE-1/ROUGE-2/ROUGE-L

As for the summarization task we applied ROUGE metrics by comparing the lead of the article with the generated output of the models. The disadvantages of these automatic evaluation metrics, and generally the problem of using the lead as summarization is that the ROUGE metric

Table 2. A sample of an article and its lead

Article:
Egyre többet hallani arról, hogy az okostelefon-gyártók olyan ujjlenyomatolvasókkal kísérleteznek, amelyek be lennének építve a kijelzőbe. Korábban sokan úgy gondolták, hogy a Galaxy S8-aknál vagy legalábbis a Galaxy Note 8-nál jelentheti be az újdonságot a Samsung, de úgy tünik, egyelőre még nem tudta ezt megoldani. A KGI neves elemzője, Ming-Chi Kuo szerint ez a funkció legkorábban a Galaxy Note 9-ben fog megjelenni, azaz még legalább egy évet várnunk kell rá. A szakember egyébként jól ismert pontos jövendöléseiről, bár elsősorban az Apple és nem a Samsung termékeire összpontosít. A Samsung már a Galaxy S8-nál üveg alá tette az ujjlenyomat-olvasót (aminek az elhelyezése amúgy nem váltott ki egyöntetü lelkesedést) , de a kijelző alá tenni azt, egyelőre még senkinek sem sikerült. Pedig ha ez megvalósulna, az sok lehetőséget nyithatna meg a gyártók előtt. Kuo szerint a Samsung már eldöntötte, hogy nem változtat az ujjlenyomat-olvasón a Galaxy Note 8-nál, viszont a Note 9-nél már valószínüleg az új megoldást mutatja be. Kuo azt is megszellőztette, hogy a dél-koreai óriás lecseréli biometrikus szkenner partnerét, a Synapticsot egy Egis nevü cégre.
Translation:
It is more and more commonly presumed that the smartphone manufacturers experiment with fingerprint readers that are built in the screen. Previously, many people thought that Samsung would announce the novelty in the case of Galaxy S8 or at least in the case of Galaxy Note 8, however, it seems that they could not solve this issue yet. According to the renowned analyst of KGI, Ming-Chi Kuo, this function will first appear first in the Galaxy Note 9, so we have to wait for it at least one more year. The specialist is well-known for his adequate predictions, however, he is more focused on Apple products rather than Samsung products. Samsung already placed the fingerprint reader under the glass in the case of Samsung Galaxy S8 (by the way this positioning did not evoke a uniform appreciation), but nobody has ever succeeded at placing it under the screen. The realization of this idea would clearly open up new opportunities in front of the manufacturers. According to Kuo, Samsung has already decided that they would not change the fingerprint reader configuration of Galaxy Note 8, but it is likely that they will introduce the new solution in Note 9. Kuo even leaked that the South Korean giant would change its biometric scanner partner, Synaptics, to another company named Egis.
Original lead:
Egy rég várt jellemző debütálását valószínüsítik a jövő évben megjelenő Galaxy Note 9-ben.
Translation:
A feature anticipated for a long time is likely to debut next year in Galaxy Note 9.

shows only how similar are the generated output and the lead. However, the function of the lead is usually to attract attention, and not necessarily to summarize. Despite the fact that these metrics could not provide a fully relevant measure, this is still regarded as the gold standard when international summarization tasks are evaluated.

In Table 2, you can see a sample. The article is about smartphone manufacturers developing new fingerprint reader technologies that are built into the screen of the devices. It is anticipated that the new fingerprint reader solution by Samsung would be introduced in Galaxy Note 9, though there is still quite some speculation around it. The lead of the article, as an attention attraction text, does not mention the concrete fingerprint function, just refers to it as ‘A feature’, but it names the Note 9 where this new feature will appear.

6. EXTRACTIVE SUMMARIZATION EXPERIMENTS AND RESULTS

In the case of extractive summarization, as a follow-up of our previous research (Yang, Perlaki & Laki 2020, 2021; Yang 2022a), huBERT, ELECTRA, ELECTRIC and the HILBERT models were fine-tuned with the BertSum tool (Liu 2019), without any hyper-parameter modification, except for the batch size. To avoid the CUDA out-of-memory error, the batch size was decreased to 20.

In our previous and recent research, the following models were fine-tuned:

- **mBERT** (Devlin et al. 2019): multilingual cased BERT model that contains Hungarian language.
- **huBERT wiki** (Nemeskey 2020a): Hungarian BERT base model trained on Hungarian Wikipedia.
- **huBERT** (Nemeskey 2021): State of the art Hungarian BERT base model. I have trained this model in this recent research.
- **HILBERT** (Feldmann et al. 2021): Hungarian BERT large model.
- **ELECTRA base** models: Hungarian ELECTRA base models. There are two variants: ELECTRA base wiki trained on Hungarian Wikipedia and ELECTRA base nytk trained on NYTK corpus.
- **ELECTRA small** models (Yang 2022a): Hungarian ELECTRA small models. There are three variants:
  - ELECTRA small 64 wiki trained on Hungarian Wikipedia with 64k size of vocabulary.
  - ELECTRA small 64 nytk trained on NYTK corpus with 64k size of vocabulary.
  - ELECTRA small 31 nytk trained on NYTK corpus with 31k size of vocabulary.
- **ELECTRIC small** models (Yang 2022a): Hungarian ELECTRIC small models. There are two variants: ELECTRA small nytk trained on NYTK corpus and ELECTRA small nytk 10% is a checkpoint at 10% of the training time of the ELECTRA small nytk model.

To fine-tune the models, the HVG corpus was used. For a better comparison, all experiments were carried out using the same train-valid-test subcorpora as in the research of (Yang et al. 2021).

In Table 3, the results of the extractive summarisation experiments can be seen. Both Recall and F-measure values are shown. huBERT and HILBERT achieved the highest performance. The results we acquired match our expectations, since ELECTRA and ELECTRIC small models have fewer parameters and ELECTRA base models were trained on smaller corpora. huBERT model was trained on the largest corpus and it is the state-of-the-art Hungarian BERT model. HILBERT model can learn more features due to its greater size. The difference between the top two models is that huBERT could gain higher F-measure, while HILBERT could gain higher Recall results. As a conclusion, it can be stated that the sentences that were chosen by the huBERT are shorter and more precise. Another notable result is that the ELECTRA and ELECTRIC small models could achieve competitive performance, despite the fact that these models have much fewer parameters and require much fewer resources.

7. ABSTRACTIVE SUMMARIZATION EXPERIMENTS AND RESULTS

In the abstractive summarization experiments, I followed up on our previous research (Yang et al. 2021; Agócs et al., 2022; Yang 2022c, b), then I tested and further fine-tuned the BME-

Table 3. Extractive summarization ROUGE Results

	Recall	F-measure
mBERT	48.58/20.12/39.42	26.38/09.98/21.16
huBERT wiki	48.86/20.45/39.60	27.21/10.51/21.82
huBERT	49.45/21.07/40.14	**27.35/10.78/21.97**
ELECTRA base wiki	48.83/20.37/39.53	26.30/10.06/21.06
ELECTRA base nytk	49.04/20.53/39.76	26.37/10.13/21.15
ELECTRA small 64 wiki	49.02/20.52/39.74	26.36/10.11/21.13
ELECTRA small 64 nytk	48.99/20.51/39.70	26.38/10.13/21.13
ELECTRA small 31 nytk	49.04/20.53/39.76	26.37/10.12/21.15
ELECTRIC small nytk 10%	49.05/20.54/39.77	26.38/10.13/21.16
ELECTRIC small nytk	49.07/20.56/39.79	26.40/10.14/21.17
HILBERT	**49.87/21.93/40.33**	27.02/10.43/21.60

TMIT/foszt2oszt model (Makrai et al. 2022). Additionally, I fine-tuned the mT5 base and the mBART-50 models for summarization tasks. I compare 14 different models on 3 different corpora for Hungarian. The main model types are the following:

- **PreSumm** models (Yang et al. 2021): there are three variants:
  - PreSumm-mBERT: multilingual cased BERT model (Devlin et al. 2019) that contains Hungarian language. Using the PreSumm tool, the model was fine-tuned for Hungarian abstractive summarization task.
  - PreSumm-huBERT: State-of-the-art Hungarian BERT base model (Nemeskey 2021). Using the PreSumm tool the model was fine-tuned for Hungarian abstractive summarization task.
  - PreSumm-HILBERT: Hungarian BERT large model (Feldmann et al. 2021). Using the PreSumm tool the model was fine-tuned for Hungarian abstractive summarization task.
- **NYTK-GPT-2** (Yang 2022b): Hungarian experimental GPT-2 model that was trained on Hungarian Wikipedia and then fine-tuned on HI corpora.
- **BART** models (Yang 2022c): There are four variants:
  - BART-base-512: Hungarian BART base model with 512 input size that was trained on Webcorpus 2.0 (Nemeskey 2020b), then it was fine-tuned for Hungarian abstractive summarization task.
  - BART-base-1,024: Hungarian BART base model with 1,024 input size that was trained on Webcorpus 2.0 (Nemeskey 2020b), then it was fine-tuned for Hungarian abstractive summarization task.
  - BART-large: Hungarian BART large model that was trained on Webcorpus 2.0 (Nemeskey 2020b), then it was fine-tuned for Hungarian abstractive summarization task.
  - BART-base-enhu: English-Hungarian bilingual BART base model that was trained on English and Hungarian Wikipedia, then it was fine-tuned for Hungarian abstractive summarization task.

- **mT5 fine** (Xue et al. 2021): Multilingual T5 base model that contains 101 languages, including Hungarian.
- **mBART-50 fine** (Tang et al. 2021): mBART-50 model which is a multilingual BART model that contains 50 languages, but **not** including Hungarian.
- **BME-TMIT/foszt2oszt** (Makrai et al. 2022) (foszt2oszt): Hungarian BERT base Encoder-Decoder transformer model for abstractive summarization. In my experiment, the model was tested directly and I have also fine-tuned (foszt2oszt fine) on my own summarization corpora.
- **-ft** models: In the case of PreSumm-mBERT (PreSumm-mBERT-tf) and BART-base-enhu (BART-base-enhu-tf), there are transfer experiments, which means, as a first step the language model was fine-tuned on CNN/Daily Mail corpora,[13] then in the second phase, the model was further fine-tuned on the Hungarian corpora.

First, multilingual models were fine-tuned on CNN/Daily Mail corpora to train the transfer models. In Table 4, the results of the fine-tuning for English can be seen. As we expected, our results could not achieve the original performance, but it can be an appropriate basis for transfer fine-tuning.

As for the evaluation of abstractive summarization models, the ROUGE (Lin 2004) F-measure metric was used. In Table 5, the results of the abstractive summarization experiments can be seen. Above the double line, the results of the previous research are shown, except for the BART-large NOL and PreSumm-mBERT-tf MARCELL values. These two measurements and the values under the double line are the results from the current experiments. As you can see in Table 5, most of my current experiments could achieve higher performance than our previous experimental works or set a new state-of-the-art.

The most notable result is that the fine-tuned mBART-50 model could gain the highest results in several tasks despite the lack of Hungarian pre-training. Another intriguing result to be emphasized is that the further fine-tuned huBERT-based foszt2foszt, which has fewer parameters, could achieve the highest performance (ROUGE-2, ROUGE-L) on HI corpus. It means the fewer parameters are outweighed by the "well-taught" Hungarian pre-trained knowledge. On NOL corpus, the fine-tuned mT5 model could gain the highest performance. These experiments can show that the different model types could achieve different performance on different data types. It is not possible to clearly determine which model architecture is the best, it is dependent on the dataset and the actual task.

In Table 6, you can see the length of leads generated by the different models. For the best comparison, HI test corpus was used. Generally, we can say that the PreSumm method generate

Table 4. Abstractive summarization results on CNN/Daily Mail corpora

PreSumm BERT original	41,72/19,39/38,76
PreSumm-mBERT	25,76/10,91/24,37
BART original	44,16/21,28/40,90
NYTK-BART-base-enhu	40,07/17,61/27,35

[13] https://github.com/abisee/cnndailymail

Table 5. Abstractive summarization results

	HI	**NOL**	**MARCELL**
PreSumm-huBERT	22,42/10,24/18,72	26,34/10,90/22,01	75,85/68,35/74,61
PreSumm-mBERT	28,34/12,40/23,45	30,56/11,57/24,99	72,99/65,38/65,38
PreSumm-mBERT-ft	27,81/11,71/22,81	30,34/10,83/24,42	76.83/69.92/75.52
BART-base-512	30.18/13.86/22.92	46,48/32,40/39,45	71,25/62,79/69,75
BART-base-1024	31,86/14,59/23,79	47,01/32,91/39,97	71,01/62,58/69,42
BART-large	30,12/13,07/22,72	38.05/21.99/30.27	70,24/60,69/68,53
BART-base-enhu	31,36/14,34/23,48	42,71/27,59/35,38	71,47/63,04/69,93
BART-base-enhu-ft	31,76/14,47/23,47	45,05/30,46/37,64	77,06/70,64/75,96
PreSumm-HILBERT	17,36/05,41/14,14	–	–
NYTK-GPT-2	23.06/06.56/15.04	–	–
foszt2oszt	27.63/09.65/19.99	28.04/09.02/18.86	14.38/04.65/13.19
foszt2oszt fine	34.89/**17.04/26.25**	47.44/31.52/38.81	74.99/67.55/74.50
mT5 fine	33.30/15.97/24.65	**54.07/40.74/46.96**	**79.45**/72.88/78.17
mBART-50 fine	**35.17**/16.46/25.61	50.56/36.59/43.16	79.03/**73.67/78.29**

Table 6. Lead length on HI corpus generated by the models

	Token (avg/median)	**Sentence (avg/median)**
Reference	25.44/23	1.39/1
PreSumm-huBERT	89.27/94	4.79/5
PreSumm-mBERT	66.44/66	4.72/4
PreSumm-mBERT-tf	65.16/65	4.10/4
PreSumm-HILBERT	92.99/99	5.61/5
BART-base-512	25.63/22	1.25/1
BART-base-1024	25.16/22	1.25/1
BART-large	19.78/18	1.16/1
BART-base-enhu	25.54/22	1.27/1
BART-base-enhu-tf	27.74/24	1.35/1
NYTK-GPT-2	59.10/57	3.00/3
foszt2oszt	22.96/22	1.00/1
foszt2oszt fine	22.28/21	1.02/1
mT5	34.10/29	1.49/1
mBART-50	23.51/21	1.28/1

Table 7. A sample of the summary generated by the PreSumm and the NYTK-GPT-2 models

Original lead:
Egy rég várt jellemző debütálását valószínüsítik a jövő évben megjelenő Galaxy Note 9-ben.
Translation:
A feature anticipated for a long time is likely to debut next year in Galaxy Note 9.
PreSumm (huBERT):
Ming-Chi Kuo, a KGI neves elemzője szerint a Samsung már a Galaxy Note 8-nál is bevezetheti az ujjlenyomat-olvasót, de egyelőre még nem tudni, mikor. A Samsung már eldöntötte, hogy nem változtat az ujjlenyomat-olvasón, viszont lecseréli a Synapticsot egy Egis nevü cégre. A szakember szerint ez sok lehetőséget nyithatna meg a gyártók előtt, ha megvalósulna az új funkció, akkor sok új lehetőséget nyitna meg a gyártó előtt. Ha ez megvalósulna, a Samsung lecseréli a Synaptitcsot egy Egisre.
Translation:
According to Ming Chi-Kuo, renowned analyst at KGI, Samsung might already introduce the fingerprint in Galaxy Note 8, but when - it is not known yet. Samsung has already decided that it won't change the fingerprint reader, but it will change Synaptics to another company called Egis. The specialist states that it would open up new opportunities in front of the manufacturers if this function was realized, then it would open up a lot of new opportunities in front of the manufacturer. If it was implemented, Samsung will change Synaptics to an Egis.
PresSumm (HILBERT):
Úgy tünik, a Samsung nem fogja bemutatni a Galaxy Note 8 ujjlenyomat-olvasóját, a Galaxy S8-at. A dél-koreai cég a Galaxy S8-cal jelentette be, hogy a készülékben már elérhető lesz az ujjlenyomat-olvasó. A Galaxy S8-at már tesztelik, de egyelőre nem tudni, mi lesz a készülék sorsa. Úgy tudni, a gyártónak van egy fontos kifogása a Galaxy S8-nál, de a gyártónak ez nem tetszik.
Translation:
It seems that Samsung won't present the fingerprint reader of Galaxy Note 8, the Galaxy S8. The South Korean company announced in the case of Galaxy S8 that the fingerprint reader will already be available in the appliance. The Galaxy S8 is already under testing, but it is not known what will be the fate of the appliance. It is known that the manufacturer has a major issue with the Galaxy S8, but the manufacturer is not happy about it.
NYTK-GPT-2:
A Samsungtól független amerikai tech-elemző cég, a KGI az egyre növekvő kijelzőgyártási lázról és a Galaxy Note 8-ak formavilágáról közölt egy érdekes cikket. Úgy tünik, a Samsung lerántotta a leplet az ujjlenyomat-olvasóról. Meg kell várni, hogy a Samsung kereskedelmi forgalomba dobja az Android legújabb megoldásait, a Note 9-et.
Translation:
KGI, an American tech-analyst company that is independent of Samsung, published an interesting article on the screen-manufacturing fever and the design of Galaxy Note 8. It seems that Samsung uncovered the fingerprint reader. We have to wait if Samsung commercially releases the latest solutions by Android, the Note 9.

very long leads, especially the HILBERT. Thus, the recall results are really high in the research of Yang et al. (2021), but the F1-scores are lower. Similarly to PreSumm scores, the NYTK-GPT-2 also generates long summaries, which is due to the identical experimental settings applied in the research by Yang (2022b), therefore, the number of generated sentences was manually set to 3. BART-based models, both Hungarian and multilingual mBART could generate similar text length as the reference. The results are similar to the foszt2oszt model. These models can learn the length of the text as well. The mT5 performance is in between that of the PreSumm and BART models.

In Tables 7–9, you can see a sample from the models used in our experiments. In Table 7, you can see a sample from PreSumm and NYTK-GPT-2 models. The general observation about PreSumm is that these summaries are long (as mentioned above), consequently, these models have

Table 8. A sample of the summary generated by the BART models

Original lead:
Egy rég várt jellemző debütálását valószínűsítik a jövő évben megjelenő Galaxy Note 9-ben.
Translation:
A feature anticipated for a long time is likely to debut next year in Galaxy Note 9.
BART-base-1024:
A Samsung egyik neves elemzője, Ming-Chi Kuo szerint legalább egy évet kell várni rá, hogy a Galaxy S8-nál is megjelenhessen ujjlenyomat-olvasó.
Translation:
According to one of the renowned analysts of Samsung, Ming-Chi Kuo, we must wait at least 1 year for the appearance of the fingerprint reader in Galaxy S8.
BART-large:
Ha igazak a híresztelések, jövőre egy új Samsung-csúcstelefon kerülhet a boltok polcaira.
Translation:
If the news statements are right, next year a new high-end Samsung smartphone can appear on the shelves.
BART-enhu:
Az ismert Ming-Chi Kuo szerint a Samsung már eldöntötte, hogy nem változtat az ujjlenyomat-olvasón.
Translation:
According to the renowned Ming-Chi Kuo, Samsung has already decided that it won’t make changes on the fingerprint reader.
BART-enhu-tf:
Ha hihetünk a pletykáknak, legkorábban a Galaxy Note 9-ben jelenik majd meg a Samsung új, ujjlenyomat-olvasós funkciója.
Translation:
Rumor says that the new fingerprint reader function of Samsung earliest will appear in Galaxy Note 9.

high recall values. The outputs are usually relevant and have only minor grammatical mistakes. The PreSumm model often generates text fragments at the end of the text, in the final application of the model (Yang et al. 2021), these fragments are removed. It is notable that the PreSumm (huBERT) is the only model that mentioned the information about “Egis” from the article.

In Table 7, you can see an example from NYTK-GPT-2 as well. The number of generated output of GPT-2 was set to three sentences. Thus, the length of output is similar to PreSumm models. As Yang described in his paper (Yang 2022b), this model is often hallucinating. For instance, in Table 7, the part of “Android legújabb megoldásait, a Note 9-et” (“the latest solutions by Android, the Note 9.”) is an uncorrect output, the article is not even about ‘Android’, and the “Note 9” is not a solution of the ‘Android’.

In Table 8, you can see the output of fine-tuned bilingual BART models. All the summaries are relevant and well-formed. The only difference is the focus of the topic. The BART-large model puts an emphasis on the appearance of the smartphone and the other models focus on the fingerprint feature. The original lead actually refers to this new fingerprint function, but it did not explicitly write about it.

Table 9. A sample of the summary generated by the models fine-tuned in this current research

Original lead:
Egy rég várt jellemző debütálását valószínűsítik a jövő évben megjelenő Galaxy Note 9-ben.
Translation:
A feature anticipated for a long time is likely to debut next year in Galaxy Note 9.
foszt2oszt fine:
A neves elemző, Ming - Chi Kuo szerint leghamarabb a Galaxy Note 9 - ben jelenik majd meg az ujjlenyomat - olvasó.
Translation:
According to the renowned analyst, Ming-Chi Kuo, the fingerprint reader will appear the earliest in Galaxy Note 9.
mT5 fine:
A KGI neves elemzője, Ming-Chi Kuo szerint a Samsung már eldöntötte, hogy nem változtat az ujjlenyomat-olvasón a Galaxy Note 8-nál, viszont a Note 9-nél már valószínüleg az új funkciót mutatja be.
Translation:
According to the renowned analyst of KGI, Ming-Chi Kuo, Samsung has already decided that it wouldn’t change the fingerprint reader in Galaxy Note 8, however, it will likely present the new function in Note 9.
mBART fine:
A KGI neves elemzője, Ming-Chi Kuo szerint a Galaxy Note 8 után a Galaxy Note 9 - ben lehet a Samsung első ujjlenyomat-olvasója.
Translation:
According to the renowned analyst of KGI, Ming Chi-Kuo, that after the Galaxy Note 8, the first fingerprint reader of Samsung will likely appear in the Note 9.

In Table 9, you can see the output of the current models with the highest performance. All three models wrote about the fingerprint feature, and all three models mentioned that this feature will appear in Note 9. This information, among BART models, it was only mentioned in the output made by BART-enhu-tf model. As a comparison to PreSumm models, none of them referred to this information. The PreSumm models wrote only about the Note 8.

Generally, we can conclude that the generated leads are mostly grammatically and syntactically correct. The main difference besides the length of the leads is the meaning that these leads are focusing on. It is a difficult question even for a human to highlight the real essence of an article. If we highlight the main message that the fingerprint feature is important and this feature will appear only in Note 9 and not in Note 8, then the foszt2foszt fine, mT5 fine, mBART fine models gave the best results.

Our future goal is to combine the advantages of the different models using algorithms like ensembling or voting (Tajti, 2020).

8. CONCLUSION

The retrieval of relevant information from written texts is a time- and labour-intensive process, therefore there is an increasing demand to optimize and automate the procedures that can quickly and efficiently create summaries of documents. In line with these attempts, more and more implementations have been made to develop neural machine learning-based solutions that can provide game-changing solutions in that direction. With the clear dominance of English language, highly efficient extractive and abstractive summarization methods have been created. However, the scarcity of summary generation methods for Hungarian motivated us to contribute to the development of novel strategies in this area. We can train our own Hungarian language models, then fine-tune them to our summarization task, or we can adapt multilingual models to Hungarian. Furthermore, a multilingual model without Hungarian knowledge can also be adapted. Our results suggest that using an implementation of the mBART model we could achieve the best outcomes, which is a somewhat surprising finding given the apparent lack of pre-training of mBART in Hungarian. Overall, most of our models could achieve state-of-the-art results in both extractive and abstractive summary generation offering a breakthrough transition from traditional summarization to a new era of artificial intelligence-supported summary generation systems.

REFERENCES

Abdel-Salam, Shehab and Ahmed, Rafea. 2022. Performance study on extractive text summarization using BERT models. Information 13(2).

Agócs, Ádám and Zijian Győző, Yang. 2022. Absztraktív összefoglaló PreSumm módszerrel (Abstractive summarisation using the PreSumm method). XVIII. Magyar Számítógépes Nyelvészeti Konferencia. 241–255.

Aharoni, Roee, Melvin, Johnson and Orhan, Firat. 2019. Massively multilingual neural machine translation. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Vol. 1 (Long and Short Papers). 3874–3884.

Akiyama, Kazuki, Akihiro, Tamura and Takashi, Ninomiya. 2021. Hie-BART: Document summarization with hierarchical BART. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Student Research Workshop. 159–165.

Baevski, Alexei, Sergey, Edunov, Yinhan, Liu, Luke, Zettlemoyer and Michael, Auli. 2019. Cloze-driven pretraining of self-attention networks. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). 5360–5369.

Bajgar, Ondrej, Rudolf, Kadlec and Jan, Kleindienst. 2017. Embracing data abundance: BookTest dataset for reading comprehension. Proceedings of the 5th International Conference on Learning Representations (ICLR 2017).

Basols, Guifré Ballester. 2021. Text summarization of online hotel reviews with sentiment analysis. M.Phil. thesis. Universitat Politècnica de Catalunya – BarcelonaTech, Barcelona.

Brown, Tom, Benjamin, Mann, Nick, Ryder, Melanie, Subbiah, Jared D, Kaplan, Prafulla, Dhariwal, Arvind, Neelakantan, Pranav, Shyam, Girish, Sastry, Amanda, Askell, Sandhini, Agarwal, Ariel, Herbert-Voss, Gretchen, Krueger, Tom, Henighan, Rewon, Child, Aditya, Ramesh, Daniel, Ziegler, Jeffrey, Wu, Clemens, Winter, Chris, Hesse, Mark, Chen, Eric, Sigler, Mateusz, Litwin, Scott, Gray, Benjamin, Chess, Jack, Clark, Christopher, Berner, Sam, McCandlish, Alec, Radford, Ilya, Sutskever and Dario, Amodei. 2020. Language models are few-shot learners. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M. F. and Lin, H. (eds.) Advances in Neural Information Processing Systems, Vol. 33. Curran Associates, Inc. 1877–1901.

Celikyilmaz, Asli, Antoine, Bosselut, Xiaodong, He and Yejin, Choi. 2018. Deep communicating agents for abstractive summarization. Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Vol. 1 (Long Papers). 1662–1675.

Chintagunta, Bharath, Namit, Katariya, Xavier, Amatriain and Anitha, Kannan. 2021. Medically aware GPT-3 as a data generator for medical dialogue summarization. Proceedings of the Second Workshop on Natural Language Processing for Medical Conversations. 66–76.

Clark, Kevin, Minh-Thang, Luong, Quoc V., Le and Christopher D., Manning 2020a. Electra: Pre-training text encoders as discriminators rather than generators. International Conference on Learning Representations.

Clark, Kevin, Minh-Thang, Luong, Quoc V., Le and Christopher D., Manning 2020c. Pre-training transformers as energy-based cloze models. In EMNLP.

Conneau, Alexis, Kartikay, Khandelwal, Naman, Goyal, Vishrav, Chaudhary, Guillaume, Wenzek, Francisco, Guzmán, Edouard, Grave, Myle, Ott, Luke, Zettlemoyer and Veselin, Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. 8440–8451.

Dehghani, Mostafa, Stephan, Gouws, Oriol, Vinyals, Jakob, Uszkoreit and Kaiser Łukasz. 2019. Universal transformers. Proceedings of the 7th International Conference on Learning Representations (ICLR 2019).

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

Farahani, Mehrdad, Mohammad, Gharachorloo and Mohammad, Manthouri. 2020. Leveraging ParsBERT and pretrained mT5 for Persian abstractive text summarization. CoRR. abs/2012.11204.

Feldmann, Ádám, Róbert, Hajdu, Balázs, Indig, Bálint, Sass, Márton, Makrai, Iván, Mittelholcz, Dávid, Halász, Zijian Győző, Yang and Tamás, Váradi. 2021. HILBERT, magyar nyelvű BERT-large modell tanítása felhő környezetben (HILBERT, the BERT-large model for Hungarian, trained in a cloud environment). XVII. Magyar Számítógépes Nyelvészeti Konferencia. 29–36.

Fuad, Ahlam and Maha, Al-Yahya. 2022. AraConv: Developing an Arabic task-oriented dialogue system using multi-lingual transformer model mT5. Applied Sciences 12(4).

Gehrmann, Sebastian, Yuntian, Deng and Alexander, Rush. 2018. Bottom-up abstractive summarization. Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. 4098–4109.

Goodfellow, Ian, Jean, Pouget-Abadie, Mehdi, Mirza, Bing, Xu, David, Warde-Farley, Sherjil, Ozair, Aaron, Courville and Yoshua, Bengio. 2014. Generative adversarial nets. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N. and Weinberger, K. Q. (eds.) Advances in Neural Information Processing Systems, Vol. 27. Curran Associates, Inc. 2672–2680.

Graff, David, Junbo, Kong, Ke, Chen and Kazuaki, Maeda. 2003. English Gigaword. Linguistic Data Consortium, Philadelphia 4(1).

Graves, Alex. 2017. Adaptive computation time for recurrent neural networks. CoRR. abs/1603.08983.

Gusev, Ilya. 2020. Dataset for automatic summarization of Russian news. In A. Filchenkov, J. Kauttonen and L. Pivovarova (eds.) Artificial Intelligence and Natural Language. Cham: Springer International Publishing. 122–134.

Gutmann, Michael and Aapo, Hyvärinen. 2010. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In Y.W. Teh and M. Titterington (eds.) Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics (Proceedings of Machine Learning Research, Vol. 9). Chia Laguna Resort, Sardinia: PMLR. 297–304.

Hasan, Tahmid, Abhik, Bhattacharjee, Md. Saiful, Islam, Kazi, Mubasshir, Yuan-Fang, Li, Yong-Bin, Kang, M. Sohel, Rahman and Rifat, Shahriyar. 2021. XL-Sum: Large-scale multilingual abstractive summarization for 44 languages. Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021. 4693–4703.

Hu, Junjie, Sebastian, Ruder, Aditya, Siddhant, Graham, Neubig, Orhan, Firat and Melvin, Johnson. 2020. XTREME: A massively multilingual multi-task benchmark for evaluating cross-lingual generalisation, Proceedings of the 37th International Conference on Machine Learning (PMLR 119). 4411–4421.

Iwasaki, Yuuki, Akihiro Yamashita, Yoko Konno, Katsushi Matsubayashi. 2020. Japanese abstractive text summarization using BERT. Advances in Science, Technology and Engineering Systems Journal 5(6). 1674–1682.

Joshi, Mandar, Danqi, Chen, Yinhan, Liu, Daniel S., Weld, Luke, Zettlemoyer and Omer, Levy. 2020. SpanBERT: Improving pre-training by representing and predicting spans. Transactions of the Association for Computational Linguistics 8(1). 64–77.

Jozefowicz, Rafal, Oriol, Vinyals, Mike, Schuster, Noam, Shazeer and Yonghui, Wu. 2016. Exploring the limits of language modeling. CoRR. abs/1602.02410.

Karpathy, Andrej, Justin, Johnson and Li, Fei-Fei. 2016. Visualizing and understanding recurrent networks. Proceedings of the 4th International Conference on Learning Representations (ICLR 2016).

Lengyelné Molnár, Tünde. 2010. Automatic abstract preparation. 10th International Conference on Information: Information Technology Role in Development. 550–561.

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

Li, Liangyou, Xin, Jiang and Qun, Liu. 2019. Pretrained language models for document-level neural machine translation. CoRR. abs/1911.03110.

Lin, Chin-Yew. 2004. ROUGE: A Package for automatic evaluation of summaries. Text summarization branches out. 74–81.

Liu, Jie, Jiaye, Wu and Xudong, Luo. 2021. Chinese judicial summarising based on short sentence extraction and GPT-2. In H. Qiu, C. Zhang, Z. Fei, M. Qiu and S.-Y. Kung (eds.) Knowledge science, engineering and management. Cham: Springer International Publishing. 376–393.

Liu, Peter J., Mohammad, Saleh, Etienne, Pot, Ben, Goodrich, Ryan, Sepassi, Lukasz, Kaiser and Noam, Shazeer. 2018. Generating Wikipedia by summarizing long sequences. Proceedings of the 6th International Conference on Learning Representations (ICLR 2018).

Liu, Yang. 2019. Fine-tune BERT for extractive summarization. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). 3548–3553.

Liu, Yang and Mirella, Lapata. 2019. Text summarization with pretrained encoders. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing. 3730–3740.

Liu, Yang, Ivan, Titov and Mirella, Lapata. 2019. Single document summarization as tree induction. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Vol. 1 (Long and Short Papers). 1745–1755.

Liu, Yinhan, Jiatao, Gu, Naman, Goyal, Xian, Li, Sergey, Edunov, Marjan, Ghazvininejad, Mike, Lewis and Luke, Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. Transactions of the Association for Computational Linguistics 8(11). 726–742.

Makrai, Márton, Ákos Máté, Tündik, Balázs, Indig and György, Szaszák. 2022. Towards abstractive summarization in Hungarian. XVIII. Magyar Számítógépes Nyelvészeti Konferencia. 505–519.

Malykh, Valentin, Daniil, Chernyavskiy and Alex, Valyukov. 2020. Summary construction strategies for headline generation in the Russian. Computational Linguistics and Intellectual Technologies: Proceedings of the International Conference (Dialogue 2020). 1–9.

Merity, Stephen, Caiming, Xiong, James, Bradbury and Richard, Socher. 2017. Pointer sentinel mixture models. Proceedings of the 5th International Conference on Learning Representations (ICLR 2017).

Miculicich, Lesly, Dhananjay, Ram, Nikolaos, Pappas and James, Henderson. 2018. Document-level neural machine translation with hierarchical attention networks. Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. 2947–2954.

Miháltz, Márton. 2010. OpinHu: Online szövegek többnyelvű véleményelemzése (OpinHu: Multilingual sentiment analysis of online texts). VII. Magyar Számítógépes Nyelvészeti Konferencia. 14–23.

Nallapati, Ramesh, Feifei, Zhai and Bowen, Zhou. 2017. SummaRuNNer: A recurrent neural network based sequence model for extractive summarization of documents. Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI'17). 3075–3081.

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

Nemeskey, Dávid Márk. 2020a. Egy emBERT próbáló feladat (An emBERT-trying task). XVI. Magyar Számítógépes Nyelvészeti Konferencia. 409–418.

Nemeskey, Dávid Márk. 2020b. Natural language processing methods for language modeling. Doctoral dissertation. Eötvös Loránd University, Budapest.

Nemeskey Dávid Márk. 2021. Introducing huBERT. XVII. Magyar Számítógépes Nyelvészeti Konferencia. 3–14.

Paulus, Romain, Caiming, Xiong and Richard, Socher. 2018. A deep reinforced model for abstractive summarization. Proceedings of the 6th International Conference on Learning Representations (ICLR 2018).

Radford, Alec, Jeff, Wu, Rewon, Child, David, Luan, Dario, Amodei and Ilya, Sutskever. 2019. Language models are unsupervised multitask learners.

Raffel, Colin, Noam, Shazeer, Adam, Roberts, Katherine, Lee, Sharan, Narang, Michael, Matena, Yanqi, Zhou, Wei, Li and Peter J, Liu. 2020a. Exploring the limits of transfer learning with a unified text-to-text transformer. Journal of Machine Learning Research 21(140). 1–67.

Raffel, Colin, Noam, Shazeer, Adam, Roberts, Katherine, Lee, Sharan, Narang, Michael, Matena, Yanqi, Zhou, Wei, Li and Peter J, Liu. 2020b. Exploring the limits of transfer learning with a unified text-to-text transformer. Journal of Machine Learning Research 21(140). 1–67.

Rajpurkar, Pranav, Jian, Zhang, Konstantin, Lopyrev and Percy, Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. 2383–2392.

Rei, Marek. 2017. Semi-supervised multitask learning for sequence labeling. Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, Vol. 1 (Long Papers). 2121–2130.

Schuster, Mike and Kaisuke, Nakajima. 2012. Japanese and Korean voice search. 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). 5149–5152.

See, Abigail, Peter J., Liu and Christopher D, Manning. 2017. Get to the point: Summarization with pointer-generator networks. Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, Vol. 1 (Long Papers). 1073–1083.

Shoeybi, Mohammad, Mostofa, Patwary, Raul, Puri, Patrick, LeGresley, Jared, Casper and Bryan, Catanzaro. 2020. Megatron-LM: Training multi-billion parameter language models using model parallelism. CoRR. abs/1909.08053.

Stankevičius, Lukas and Mantas, Lukoševičius. 2021. Generating abstractive summaries of Lithuanian news articles using a transformer model. In A. Lopata, D. Gudonienė and R. Butkienė (eds.) Information and Software Technologies. Cham: Springer International Publishing. 341–352.

Tajti, T. 2020. New voting functions for neural network algorithms. Annales Mathematicae et Informaticae 229–242. https://doi.org/10.33039/ami.2020.10.003.

Tang, Yuqing, Chau, Tran, Xian, Li, Peng-Jen, Chen, Naman, Goyal, Vishrav, Chaudhary, Jiatao, Gu and Angela, Fan. 2020. Multilingual translation with extensible multilingual pretraining and finetuning. CoRR. abs/2008.00401.

Tang, Yuqing, Chau, Tran, Xian, Li, Peng-Jen, Chen, Naman, Goyal, Vishrav, Chaudhary, Jiatao, Gu and Angela, Fan. 2021. Multilingual translation from denoising pre-training. Findings of the Association for Computational Linguistics (ACL-IJCNLP 2021). 3450–3466.

Tretyak, Vladislav and Denis, Stepanov. 2020. Combination of abstractive and extractive approaches for summarization of long scientific texts. CoRR. abs/2006.05354.

Vaswani, Ashish, Noam, Shazeer, Niki, Parmar, Jakob, Uszkoreit, Llion, Jones, Aidan N, Gomez, Ł ukasz, Kaiser and Illia, Polosukhin. 2017. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan and R. Garnett (eds.) Advances in Neural Information Processing Systems, Vol. 30. Curran Associates, Inc. 5998–6008.

Váradi, Tamás, Svetla, Koeva, Martin, Yamalov, Marko, Tadić, Bálint, Sass, Bartłomiej, Nitoń, Maciej, Ogrodniczuk, Piotr, Pęzik, Verginica, Barbu Mititelu, Radu, Ion, Elena, Irimia, Maria, Mitrofan, Vasile, Păiș, Dan, Tufiș, Radovan, Garabík, Simon, Krek, Andraz, Repar, Matjaž, Rihtar and Janez, Brank. 2020. The Marcell Legislative Corpus. Proceedings of the 12th Language Resources and Evaluation Conference. 3761–3768.

Wang, Alex, Amanpreet, Singh, Julian, Michael, Felix, Hill, Omer, Levy and Samuel, Bowman. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP. 353–355.

Wenzek, Guillaume, Marie-Anne, Lachaux, Alexis, Conneau, Vishrav, Chaudhary, Francisco, Guzmán, Armand, Joulin and Edouard, Grave. 2020. CCNet: Extracting high quality monolingual datasets from web crawl data. Proceedings of the 12th Language Resources and Evaluation Conference. 4003–4012.

Xue, Linting, Noah, Constant, Adam, Roberts, Mihir, Kale, Rami, Al-Rfou, Aditya, Siddhant, Aditya, Barua and Colin, Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. 483–498.

Yang, Zhilin, Zihang, Dai, Yiming, Yang, Jaime, Carbonell, Russ R, Salakhutdinov and Quoc V, Le. 2019. XLNet: Generalized autoregressive pretraining for language understanding. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d' Alché-Buc, E. Fox and R. Garnett (eds.) Advances in Neural Information Processing Systems, Vol. 32. Curran Associates, Inc. 5753–5763.

Yang, Zijian Győző. 2022a. A kis HIL-ELECTRA, HIL-ELECTRIC és HIL-RoBERTa: Magyar kísérleti nyelvi modellek tanítása kevés erőforrással (The small HIL-ELECTRA, HIL-ELECTRIC, and HIL-RoBERTa: Training Hungarian experimental language models with low resources). XVIII. Magyar Számítógépes Nyelvészeti Konferencia. 603–617.

Yang, Zijian Győző. 2022b. „Az invazív medvék nem tolerálják a suzukis agressziót": Magyar GPT-2 kísérleti modell ("The invasive bears do not tolerate the aggression of Suzuki drivers": A Hungarian GPT-2 experimental model). XVIII. Magyar Számítógépes Nyelvészeti Konferencia. 463–476.

Yang, Zijian Győző. 2022c. BARTerezzünk! Messze, messze, messze a világtól, BART kísérleti modellek magyar nyelvre (Let's BARTer! Far, far, far away from the world, BART experimental models for Hungarian). XVIII. Magyar Számítógépes Nyelvészeti Konferencia. 15–28.

Yang, Zijian Győző, Ádám, Agócs, Gábor, Kusper and Tamás, Váradi. 2021. Abstractive text summarization for Hungarian. Annales Mathematicae et Informaticae 53. 299–316.

Yang, Zijian Győző, Attila, Perlaki and László János, Laki. 2020. Automatikus összefoglaló generálás magyar nyelvre BERT modellel (Automatic summary generation for Hungarian language using the BERT model). XVI. Magyar Számítógépes Nyelvészeti Konferencia. 343–354.

Yang, Zijian Győző and Tamáás, Váradi. 2021. Training language models with low resources: RoBERTa, BART and ELECTRA experimental models for Hungarian. Proceedings of 12th IEEE International Conference on Cognitive Infocommunications (CogInfoCom 2021). 279–285.

Zellers, Rowan, Ari, Holtzman, Hannah, Rashkin, Yonatan, Bisk, Ali, Farhadi, Franziska, Roesner and Yejin, Choi. 2019. Defending against neural fake news. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d' Alché-Buc, E. Fox and R. Garnett (eds.) Advances in Neural Information Processing Systems, Vol. 32. Curran Associates, Inc.

Zhang, Jingqing, Yao, Zhao, Mohammad, Saleh and Peter, Liu. 2020. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. Proceedings of the Thirty-seventh International Conference on Machine Learning.

Zhang, Xingxing, Mirella, Lapata, Furu, Wei and Ming, Zhou. 2018. Neural latent extractive document summarization. Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. 779–784.

Zhou, Qingyu, Nan, Yang, Furu, Wei, Shaohan, Huang, Ming, Zhou and Tiejun, Zhao. 2018. Neural document summarization by jointly learning to score and select sentences. Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, Vol. 1 (Long Papers). 654–663.