IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 3 February 2024, accepted 26 February 2024, date of publication 29 February 2024, date of current version 8 March 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3371689

RESEARCH ARTICLE

CzeGPT-2–Training New Model for Czech Generative Text Processing Evaluated With the Summarization Task

ADAM HÁJEK AND ALEŠ HORÁK
Natural Language Processing Centre, Faculty of Informatics, Masaryk University, 602 00 Brno, Czech Republic

Corresponding author: Aleš Horák (hales@fi.muni.cz)

This work was supported in part by the Ministry of Education of the Czech Republic (CR) within the LINDAT-CLARIAH-CZ Project under Grant LM2023062; and in part by the Ministry of Education, Youth and Sports of the CR through the e-INFRA CZ Project under Grant 90254.

ABSTRACT Automatic text summarization (ATS), alongside neural machine translation or question answering, is one of the leading tasks in Natural Language Processing (NLP). In recent years, ATS has experienced significant development, especially in the English NLP world. Modern approaches are mainly based on the versatile Transformer architecture proposed by Vaswani et al. in 2017, which has revolutionized the field, and was later tuned and adjusted to various needs of different tasks. Non-mainstream languages, with Czech taken as a representative, on the other hand, are a little bit behind these efforts and tend to use lighter or heuristic methods. With the new CzeGPT-2 model and abstractive summarizer, we would like to take a step forward detailing the process of training a GPT-2 generative transformer model for a new language with a comprehensive evaluation of the task of Czech summarization and pointing out the benefits of this approach. We also present an in-depth analysis of the errors in generated summaries, allowing to locate the model's weak spots.

INDEX TERMS Czech, GPT-2, large language model, model evaluation, model training, summarization.

I. INTRODUCTION

Text summarization (TS) is a process of shortening a text document while maintaining as much information as possible. The goal of TS is to get rid of nonessential content and condense the crucial points of a given text into an easily accessible form. Compressing written information is an essential step towards efficient work with textual data [1].

Abstractive text summarization (ATS) groups those automated text summarization (TS) strategies that usually implement semantic methods and language generation techniques to create a shorter re-phrased summary, an *abstract*, from scratch in a similar manner as humans approach the task [2]. **Extractive text summarization** instead tries to choose the most representative sentences from the input text and return them as a so-called extract. Although ATS shows less stable results when compared to the extractive approach, recent progress in text generation increased the abstract quality, and the summaries generally tend to be more human-like [3], [4].

The current state-of-the-art results in the summarization task are achieved almost exclusively by Transformer-based language models [1], [5]. This paper presents a detailed description of the process of training a new GPT-2 generative model [6], [7], denoted as CzeGPT-2, for Czech as a representative of non-mainstream languages. The model is evaluated with the task of abstractive text summarization using the standard $\text{ROUGE}_{\text{RAW}}$ metric and the Czech benchmark dataset named SumeCzech [8] consisting of one million newspaper articles.[1] CzeGPT-2 significantly surpasses the SumeCzech published baselines and its results are comparable to those of the 4-times larger multilingual pretrained mBART large model [9] with state-of-the-art results here. We also incorporate manual error analysis that is more subjective than the conventional $\text{ROUGE}_{\text{RAW}}$ but helps

The associate editor coordinating the review of this manuscript and approving it for publication was Vlad Diaconita.

[1] distributed under MPL 2.0 license.

us better understand the model behavior and the mechanism of mistake generation.

It is to be noted that our task here is not to compare with the largest current models such as GPT-4 by OpenAI [10] or the Gemini model by Google [11], due to the massive difference in computational costs between these cloud giant models and the presented in-house freely distributed model. By thorough evaluation, we show that for a new language, it is often sufficient to train a new model with fewer parameters than to rely purely on cloud-based commercial systems.

The research highlights of the presented article are as follows:

- Detailed description of the process of training a new freely available GPT-2 language model for a language without freely accessible large coverage language model.
- In the summarization task evaluation, the presented CzeGPT-2 model results are comparable with 4-times larger multilingual model, resulting in savings in GPU and power usage.
- Human annotations of the summarization results offer detailed error analysis of the model capabilities and reveal inadequacies of the $\text{ROUGE}_{\text{RAW}}$ metric.

The organization of the article starts with a brief summary of related works in the next section, followed by detailed specification of the methods used, including the architecture, the model with metrics employed, and a thorough description of the data processing and the training process. In Section IV, we present the evaluation with an elaborate error analysis supported by human evaluation with examples. The article concludes with an extended discussion in Section V and the Conclusions.

II. RELATED WORKS

In this overview, we concentrate on the process of training a GPT-2 generative transformer model for a new language with Czech taken as a representative. Researchers have made just a few attempts to create a Czech abstractive summarizer so far. This may be because previous methods were unable to provide satisfactory results even for widely used, morphologically less rich languages, such as English [12], [13].

A. SUMECZECH

The first big step for Czech abstractive summarization came in May 2018 with the SumeCzech summarization dataset [8]. The dataset paper also introduced an innovative evaluation system – $\text{ROUGE}_{\text{RAW}}$ – which is more suitable for free-word-order languages than the original ROUGE metric. The proposed dataset comprises about one million newspaper articles scraped from Czech news websites. Each article consists of a headline, an abstract of several sentences, and a full text, which implies three summarization tasks: *text* → *headline*, *text* → *abstract*, *abstract* → *headline*.

SumeCzech was accompanied by three unsupervised and three supervised extractive summarization techniques and three abstractive models (one for each task). The abstractive summarizers were the first attempt to train a Transformer model (the vanilla neural machine translation architecture) for Czech summarization. The performance, though, did not exceed the extractive methods.

B. NAMED ENTITIES

Following the SumeCzech methods, in April 2021, a research group from CTU in Prague came up with the idea of integrating information about the presence of named entities into the summarization process [14]. The authors tried to improve the results on a one-sentence summary task (*text* → *headline*) and proved that summaries with a higher number of named entities are more relevant since the NEs carry a significant amount of information. The authors used a recurrent neural network architecture and easily outperformed the SumeCzech baselines on the headline generation task.

C. FINE-TUNED MBART MODEL

In May 2022, M. Krotil from CTU Prague published experimental results [15] of fine-tuning the *mBART-large* [9] multilingual model with the SumeCzech dataset and a private dataset of the Czech News Center of about 750,000 documents. The large pretrained model significantly improved the precision of the generated summaries, although, interestingly, the model recall did not reach some of the baseline values.

III. METHODS

The training of Transformers for Czech has so far been a matter of only a few projects [8], [16], [17], [18], [19], [20]. To our knowledge, no project has yet pre-trained an autoregressive[2] decoder-only Transformer, which is very important for numerous tasks based on text generation.

A. ARCHITECTURE

The decoder-only Transformer is often represented by the GPT models family that Radford et al. have been proposing since 2018 [6], [22], [23]. The only significant architectural difference across the three GPT generations is the increasing size of the models. In principle, the topology of the networks has not changed.

Input preparation: Before feeding an input sequence into the Transformer, we must tokenize the text into words and subwords. Then, we replace the tokens with their assigned token IDs, which the model substitutes for the corresponding *embeddings*. Embedding is a fixed-sized trainable vector representation of a token within the model.

The Transformer, from its nature, does not recognize the order of the input tokens, which is crucial for language modeling. We provide the information using the *positional encoding vectors* that correspond to each position of the input.

[2]Autoregressive model generates the next word according to the input – after each step, the input is updated with the latest output, and the cycle continues [21].

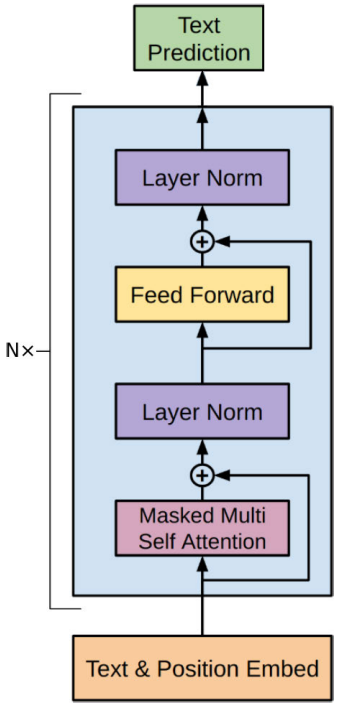

FIGURE 1. The Transformer decoder architecture employed [22].

The positional encodings are summed with the input token embeddings, and the result is sent to the network.

1) DECODER

As the name suggests, the decoder-only Transformers utilize only the decoder half of the Transformer encoder-decoder architecture. A decoder is composed of a **stack of decoder blocks** followed by a text prediction block working over the model vocabulary (see Figure 1 and the *Tokenizer* section below for details). The text prediction block consists of a **linear** and a **softmax layer**.

The objective of the decoder stack is to generate an output sequence autoregressively based on the input and the previously generated tokens. The decoder block comprises *a (masked multi) self-attention layer*, which ensures that the model does not incorporate the information about future tokens into the prediction during training, *normalization layers*, and *feed-forward layers* that share weights on every position within the decoder but are independent across the blocks.

The **linear** and **softmax layers** create a probabilistic distribution over the token IDs. Based on the distribution, we choose which token will be appended to the forming output sequence. For the generation to work well, we need to introduce a certain level of randomness. Too high as well as too low randomness deteriorates the final result. Different hyperparameters such as *top-k* or *top-p* are used to balance the behavior [24].

B. MODEL

As a foundation for CzeGPT-2, we have chosen the GPT-2 small model with 117 million trainable parameters. That means we have 1024 tokens long input/output sequence, embeddings of size 768, 12 decoder blocks, and 12 attention heads per block. The model is both reasonably large for the main summarization task and small enough to run on weaker GPUs or even CPUs.

The implementation was supported by open-source libraries with a robust training environment[3], [4].

1) TOKENIZER

GPT-2 uses a Byte-level byte pair encoding (BBPE [25]) tokenizer that has to be trained on textual data first. CzeGPT-2 tokenizer was trained on the full plain-text pre-training dataset (see Section III-D) using the Hugging Face Tokenizers library.[5] The vocabulary size was set to 50 257, which corresponds to 256 bytes in the initial alphabet, 50 000 available slots for learning, and one end-of-document token. The chosen vocabulary size is usually regarded as sufficient [26] to cover all frequent words and word prefixes and suffixes. In case of CzeGPT-2, the texts used for tokenizer training contain Czech texts only which improves the model's capabilities for processing new texts in this language.

C. METRICS

Measuring the quality of a pre-trained language model is not easy. Usually, the best choice is to select a downstream task and see how the model performs, but in the case of *summarization*, the evaluation is yet again quite challenging.

If we want to state a single quality score, the most common metrics for generative language models are *perplexity* and *accuracy*. And for the free-word-order language summarization task, the $\text{ROUGE}_{\text{RAW}}$ [8] is the current standard.

1) PERPLEXITY

Perplexity is the benchmark metric for autoregressive models that predict a token based only on the preceding sequence. We can calculate the perplexity either by cross-entropy (H) as 2^H or directly by the following formula. Having predicted a sequence $X = (x_0, x_1, \ldots, x_t)$, the perplexity of X is computed as:

$$\text{PPL}(X) = \exp\left\{-\frac{1}{t}\sum_{i=1}^{t}\log p_\theta(x_i|x_{<i})\right\}$$

where $\log p_\theta(x_i|x_{<i})$ is the log-likelihood of the i-th token with respect to the tokens $x_{<i}$ [27].

The metric intuitively illustrates the uncertainty of the language model when predicting the next token. We can understand it as an expression of the model's ability to predict a token from a fixed vocabulary stably. This mainly means that the tokenization process directly affects perplexity, which we should take into account when comparing different models, especially for different languages. Otherwise, when

[3] https://huggingface.co/transformers/model_doc/gpt2.html
[4] https://docs.fast.ai/
[5] https://github.com/huggingface/tokenizers

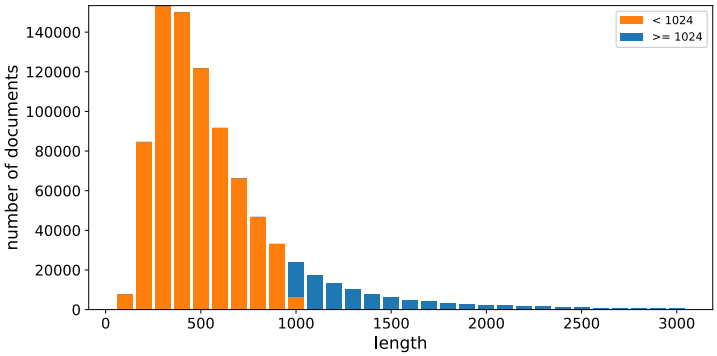

FIGURE 2. Data length distribution of the train set; all lengths were rounded down to hundreds. Orange bars denote suitable data points. Less than 0.6% of data are longer than 3000 tokens and are not displayed.

using the same tokenizer, the lower the perplexity goes, the better [28], [29].

2) ACCURACY

Along with perplexity, the value of accuracy is often stated. This metric expresses the share of correctly predicted tokens in the output sequence [30].

Even though the metric has its positives as a straightforward interpretability or given range of values, it does not describe the capabilities of the language model thoroughly since it does not consider predicted probabilities for other tokens than those included in the output [28].

3) $\text{ROUGE}_{\text{RAW}}$

The original ROUGE [31], [32] is an English-specific set of metrics and a software package that measures the similarity between generated and target summaries according to overlaps between them. The technique is based on English stemming, stop-words, and synonyms, where specifically the stemming part is too aggressive for morphologically rich languages, and stop-words and synonyms condition the metric results on extra data for each new language.

The $\text{ROUGE}_{\text{RAW}}$ metrics proposed with the SumeCzech dataset do not include any additional language-dependent steps, so they are language-agnostic [8].

There are several types of the $\text{ROUGE}_{\text{RAW}}$ metric depending on what overlaps we compute. Usually, it is either the overlap of *n-grams* (groups of n consecutive words – ROUGE-N) or the *longest common subsequence* (ROUGE-L), but other versions are available too.

The suggested variants for Czech summarization in SumeCzech are $\text{ROUGE}_{\text{RAW}}$-1, $\text{ROUGE}_{\text{RAW}}$-2, and $\text{ROUGE}_{\text{RAW}}$-L. Each variant is evaluated via its Precision, Recall, and F1-score, which support detailed interpretation. The F1-score as the harmonic mean of Precision and Recall is the most indicative of the three since it is robust against varying lengths of the summaries.

D. DATA AND TRAINING

The summarizer training procedure comprises two steps. First, we pre-train the model on unlabeled data with a broad domain to improve the world knowledge of the model. Next, we fine-tune the neural network on an annotated dataset for the abstractive summarization task.

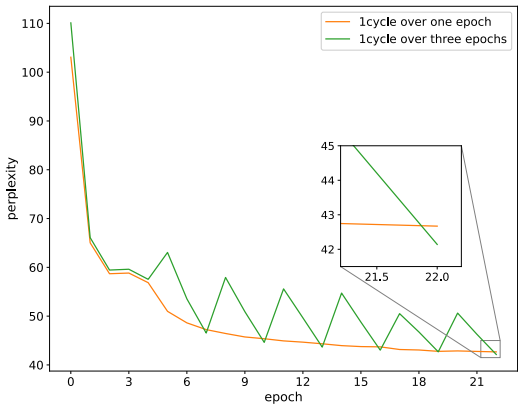

FIGURE 3. GPT-2 pre-training with different 1cycle strategies.

1) INITIALIZATION

Inspired by Pierre Guillou's experiment with Portuguese [33], we have evaluated several techniques for the model's embedding vector initialization. This kind of approach often accelerates the training of neural networks. We tried to map pre-trained embeddings of common tokens from the English GPT-2 vocabulary or an initialization using *FastText* [34] model embeddings trained specifically for this purpose on 5 GB plain-text corpus. None of these techniques significantly improved the pre-training speed.

2) PRE-TRAINING

The first phase of training CzeGPT-2 should provide a general overview of the Czech language to the model, especially syntactic and semantic relationships between words.

The CzeGPT-2 pre-training text dataset was based on the largest Czech corpus *csTenTen17* [35], [36]. The dataset is composed of Czech documents crawled from the internet, including Czech Wikipedia. During post-processing, the corpus was deduplicated, and other languages were filtered

TABLE 1. Comparison of the CzeGPT-2 based summarizer, and the introduced SumeCzech approaches on the *test set* and the *out-of-domain test set* (details in Section II). The task is *text → abstract*, and the scores denote Precision/Recall/F1-score. The best results are in ***bold italics*** and the second best in **bold** font.

Method		test set ROUGE$_{RAW}$-1	ROUGE$_{RAW}$-2	ROUGE$_{RAW}$-L
CzeGPT-2	117 M pars	**18.0/18.7/17.8**	**3.5/3.7/3.5**	**12.6/13.3/12.5**
First		13.1/17.9/14.4	1.9/2.8/2.1	8.8/12.0/9.6
TextRank		11.1/***20.8***/13.8	1.6/3.1/2.0	7.1/***13.4***/8.9
Tensor2Tensor		13.2/10.5/11.3	1.2/0.9/1.0	10.2/8.1/8.7
mT5-small$_{SumeCzech}$	300 M pars	12.4/17.6/14.1	2.0/2.8/2.3	9.5/**13.3**/10.7
mBART$_{SumeCzech}$	610 M pars	***22.9***/16.0/***18.2***	***5.7/4.0/4.6***	***16.9***/11.9/***13.5***

Method		out-of-domain set ROUGE$_{RAW}$-1	ROUGE$_{RAW}$-2	ROUGE$_{RAW}$-L
CzeGPT-2	117 M pars	**16.2/18.5/16.7**	**3.1**/3.7/**3.2**	**11.5/13.3/11.9**
First		11.1/17.1/12.7	1.6/2.7/1.9	7.6/11.7/8.7
TextRank		9.8/***19.9***/12.5	1.5/3.3/2.0	6.6/**13.3**/8.4
Tensor2Tensor		12.5/9.4/10.3	0.8/0.6/0.6	9.8/7.5/8.1
mT5-small$_{SumeCzech}$	300 M pars	12.0/***19.9***/14.3	2.3/**3.9**/2.8	9.2/***15.1***/10.9
mBART$_{SumeCzech}$	610 M pars	***23.0***/15.6/***17.9***	***6.1/4.2/4.8***	***17.1***/11.6/***13.3***

TABLE 2. Comparison of the CzeGPT-2 summarizer with approaches described in Section II on *full text → headline* task on SumeCzech *test set* and *out-of-domain test set*. The numbers denote Precision/Recall/F1-score. The best results are in ***bold italics*** and the second best in **bold** font.

Method		test set ROUGE$_{RAW}$-1	ROUGE$_{RAW}$-2	ROUGE$_{RAW}$-L
CzeGPT-2	117 M pars	**17.3/17.0/16.7**	**4.4/4.3/4.2**	**15.5/15.2/14.9**
First		7.4/13.5/8.9	1.1/2.2/1.3	6.5/11.7/7.7
TextRank		6.0/16.5/8.3	0.8/2.3/1.1	5.0/13.8/6.9
Tensor2Tensor		8.8/7.0/7.5	0.8/0.6/0.7	8.1/6.5/7.0
NE Density		6.6/10.7/7.3	0.8/1.4/0.9	5.9/9.4/6.4
Seq2Seq		16.1/14.1/14.6	2.5/2.1/2.2	14.6/12.8/13.2
Seq2Seq$_{NER}$		16.2/14.1/14.7	2.5/2.1/2.2	14.7/12.8/13.3
mT5-small$_{SumeCzech}$	300 M pars	13.3/15.7/14.0	3.0/3.5/3.1	12.2/14.3/12.8
mBART$_{SumeCzech}$	610 M pars	***19.4/17.1/17.7***	***6.1/5.4/5.5***	***17.7/15.6/16.1***

Method		out-of-domain set ROUGE$_{RAW}$-1	ROUGE$_{RAW}$-2	ROUGE$_{RAW}$-L
CzeGPT-2	117 M pars	**17.9/17.6/17.2**	**5.9/5.7/5.5**	**16.4/16.2/15.8**
First		6.7/13.6/8.3	1.3/2.8/1.6	5.9/12.0/7.4
TextRank		5.8/16.9/8.1	1.1/3.4/1.5	5.0/14.5/6.9
Tensor2Tensor		6.3/5.1/5.5	0.5/0.4/0.4	5.9/4.8/5.1
NE Density		6.3/11.4/7.1	1.3/2.3/1.4	5.7/10.2/6.3
Seq2Seq		13.1/11.8/12.0	2.0/1.7/1.8	12.1/11.0/11.2
Seq2Seq$_{NER}$		16.2/14.1/14.7	2.5/2.1/2.2	14.7/12.8/13.3
mT5-small$_{SumeCzech}$	300 M pars	14.1/16.3/14.6	4.3/5.1/4.5	13.1/15.1/13.6
mBART$_{SumeCzech}$	610 M pars	***20.8/18.8/19.2***	***8.0/7.2/7.3***	***19.2/17.4/17.7***

out. Overall, the dataset[6] comprises 12.5 billion tokens. For the pre-training itself, we used a 5 GB random slice from the corpus. We split the data to train/test/validation sets with the ratio of 90 : 5 : 5.

The model was pre-trained for 135 hours on an A100 GPU card using the `fastai` library. The batch size was set to 8, which was the highest possible to fit into the GPU memory. Using the `fastai` *lr_finder* we chose the initial learning rate value of 0.002 and used the 1cycle policy suggested by Leslie N. Smith in [37]. We tried to span the 1cycle policy over one and three epochs, but it had no significant impact on the training quality (see Figure 3). After 104 hours of training, when the validation perplexity began to rise, we dropped the learning rate to 0.001 and were able to decrease the perplexity further until the local minimum with the final perplexity of 42.14.

[6]The corpus data is available at http://hdl.handle.net/11234/1-4835

3) FINE-TUNING

The CzeGPT-2 fine-tuning was performed with the SumeCzech summarization dataset [8] that is composed of about one million newspaper articles divided into train,

validation, test, and out-of-domain (OOD) test sets. The OOD test set is a cluster of 4.5% in size extracted with the K-Means algorithm, and the rest of the data was divided into train/test/validation using an 86.5 : 4.5 : 4.5 split.

Because the CzeGPT-2 model has a fixed-sized input layer that can take only 1024 tokens, we need all the data points to have a maximum article length and abstract length together with 1023 tokens, with one token left for a separator. Statistics of the content lengths reveal that 87.9% of the data meets these requirements (see Figure 2). What we also discovered is that the length distribution is nearly equal across all splits. This means that the evaluation is not negatively impacted by a different length of test inputs compared to the training data.

The fine-tuning process was implemented using the Hugging Face Transformers[7] library. We trained the model for 100 hours (15 epochs) on an A100 GPU, but the crucial drop of evaluation loss happened in the first six epochs. This time, the maximal possible batch size was 4, so we increased it with *gradient accumulation* to 64. We did not use the 1cycle policy; the learning rate was maintained by the Adam optimizer.

From the general pre-trained model, two summarizers were fine-tuned – one for the *text* → *abstract* task and one for the *text* → *headline* task. The training objective is still the same – predicting the next token in a sequence according to the preceding context. The training strategy is to give our model the text of an article as context and teach it to generate the abstract (or the headline, respectively).

To separate the two parts of the input, a special token `<|sep|>` was added to the tokenizer vocabulary. This token is trained to inform the model that the context sequence has ended, and now it is the time to generate the summary.

In the case of fine-tuning, we aim not to model the language in general but more specifically to the final task. Therefore, we want to punish the network only for its mistakes while summarizing, not generating the rest of the article. For this purpose, we mask the target labels so only the summary tokens and separators contribute to the error function.

Also, for the abstract generator, we decided not to train the embedding of the `<|endoftext|>` token, which allows us to create abstracts of arbitrary length. For the headline generation, on the other hand, it is more beneficial to let the model decide when to stop and be independent of punctuation.

Later, based on tuning on the validation set, we decided to generate three-sentence abstracts and use *top-k* of 50 and *top-p* of 0.5.

IV. RESULTS

For evaluating the CzeGPT-2 summarizer, two approaches have been used. The first is the automatic evaluation using the $\text{ROUGE}_{\text{RAW}}$ software package provided by authors of the SumeCzech, and the latter is a detailed error analysis produced manually by human annotators on a subset of the data.

A. $\text{ROUGE}_{\text{RAW}}$ EVALUATION

Apart from the standard test set, SumeCzech provides an out-of-domain test set composed of articles with a different topic than the rest of the partitions. Since the models cannot accept inputs longer than 1024 tokens, we had to filter approximately 12% of the data for abstract and 9% of the data for headline generation.

The $\text{ROUGE}_{\text{RAW}}$ results of CzeGPT-2 with the entire test set and the out-of-domain test set compared to the approaches of [8],[8] the named-entities method [14], the mT5-small multilingual model by Google Research [38] fine-tuned with SumeCzech,[9] and the fine-tuned mBART model [15], mentioned in Section II, are in Tables 1 and 2.

1) TEXT → ABSTRACT

In the abstract generation task, the CzeGPT-2 model outperformed all the SumeCzech baselines including the fine-tuned mT5-small model and achieved stable results across precision, recall, and F1-score. This stability is crucial because it means that the length of the summary does not bias the score – short summaries tend to have larger precision with lower recall, and longer summaries the other way around. F1-score is robust against this behavior. This may be the reason why both the *TextRank* summarizer and CzeGPT-2 have reached higher recall of $\text{ROUGE}_{\text{RAW}}$-1 and $\text{ROUGE}_{\text{RAW}}$-L than the state-of-the-art mBART large model (see Table 1).

With the OOD test set, the CzeGPT-2 model moves the bar above the baselines for almost all metrics, too. The deterioration of the result is noticeable in the unknown domain, but the model apparently generalizes without issues.

2) TEXT → HEADLINE

In the headline generation task, the CzeGPT-2 model also did a very good job. It beats the Named Entity RNN summarizers and beats all the compared state-of-the-art results except the pretrained mBART large model for all metrics on both test and OOD test sets (see Table 2). Examples of the gold and generated headlines are presented in Table 3.

B. ERROR ANALYSIS

Even though $\text{ROUGE}_{\text{RAW}}$ is the best metric for summarization we have right now, it is far from ideal. $\text{ROUGE}_{\text{RAW}}$ only tells us if the model used the same words in some form as the ground truth. Unfortunately, with the advent of abstractive summarizers, factual and grammatical errors occur in the output, which ROUGE cannot reveal. With the advancements in the largest current models such as GPT-4 [10] or Gemini [11], the discourse coherence can be assessed by processing the summarization results by

[7]https://github.com/huggingface/transformers

[8]In the case of SumeCzech, we chose only the best models.

[9]The mT5-small model has 300 million learnable parameters, i.e. twice as much as CzeGPT-2.

TABLE 3. Examples of golden and generated headlines. We can see the practical limitations of the $ROUGE_{RAW}$ metric – the first sentence has the $ROUGE_{RAW}$-1 F1-score of 0 but it is completely correct, and the meaning agrees with the golden headline. On the other hand, the second example has the $ROUGE_{RAW}$-1 F1-score of 0.55 and the generated headline has almost opposite meaning than the ground truth.

Gold 1
Moderní trendy ve vytápění *(Modern trends in heating)*
Generated by CzeGPT-2
Pohodlné a příjemné teplo pro váš dům *(Comfortable and pleasant warmth for your home)*

Gold 2
Paroubek: Akci CzechTek jsem podcenil *(Paroubek: I underestimated the CzechTek event)*
Generated by CzeGPT-2
Paroubek: Policie podcenila CzechTek *(Paroubek: Police underestimated CzechTek)*

TABLE 4. The *Mapping* dimension of the summarization errors [39].

	Mapping
Omission	Copying words from an article sentence but omitting necessary words or phrases.
Wrong combination	Copying words or phrases from multiple article sentences and combining them into an erroneous sentence.
Fabrication	Introducing one or multiple new words or phrases that cause an error.
Lack of re-writing	Failing to adequately re-write sentences, e.g., by not replacing referential expressions with their original antecedents in the text. When the antecedents are not present in the preceding summary context, this causes an error.

TABLE 5. The *Meaning* dimension of the summarization errors divided into the *Malformed* and *Misleading* subcategories [39].

	Malformed
Ungrammatical	A sentence that is syntactically unnatural and would not be uttered by a competent speaker. Syntactically malformed.
Semantically implausible	A sentence that is semantically unnatural and would not be uttered by a competent speaker. Nonsensical due to semantic errors.
No meaning can be inferred	A sentence that is grammatically correct but to which no meaning can be assigned, even after accommodation.
	Misleading
Meaning changed, not entailed	In the summary context, the semantic content assigned to a sentence is not entailed by the original article.
Meaning changed, contradiction	In the summary context, the semantic content assigned to a sentence is in contradiction to the article.
Pragmatic meaning changed	In the summary context, the sentence gains a pragmatic meaning not present in the original article. Or, a pragmatic meaning present in the article is lost.

these models [40]. A disadvantage of this approach lies in the increased evaluation price and still reduced capabilities in judging subtle factual errors when compared to human evaluation.

To see how good summaries our model actually generates, what are the most frequent types of errors and how these errors arise, we decided to perform a manual annotation of a subset of the generated summaries and classify the mistakes.

1) METHODOLOGY

We use the methodology suggested by [39] that assigns each error a category in two dimensions denoted as *mapping* and *meaning*.

Mapping describes the surface level – what mechanism the model used to compose the erroneous sentence, what words or phrases it combined or omitted. This dimension reveals the source of a mistake and can help us avoid it. The four mapping categories and their definitions are listed in Table 4.

The second dimension, *meaning*, focuses on the *effect* of the error. It tells the impact of the error on the syntax, semantics, and meaning of the sentence. The Meaning dimension is divided into two subdimensions – *malformed* and *misleading* – and each of them has three categories (see Table 5). The annotators choose only one of these six options for each error.

2) COURSE OF THE ANALYSIS

To cover all aspects of the summarization dataset in an annotation subset, we have identified four groups of the generated data – the abstracts in the test set, the abstracts in the OOD set, the headlines in the test set, and the headlines in the OOD set. From each of these groups, we took the best 15 and the worst 15 summaries for the purpose of the annotation of errors. The selection was made based on the $ROUGE_{RAW}$-1 F1-score, so we can also inspect which

TABLE 6. An example of an article and its golden and generated summary (abstract). The first three sentences of the article are also copied to the golden summary (which is a known problem in the SumeCzech dataset). The first sentence was wrongly rephrased by the CzeGPT-2, and an erroneous Semantically implausible sentence was generated.

Article
Generální ředitel mobilního operátora Eurotel Terrence Valeski by mohl zasednout v představenstvu Českého Telecomu. Podle dobře informovaného zdroje mu to nabídl předseda představenstva a generální ředitel Telecomu Gabriel Berdár a Valeski nabídku přijal. "Je pravda, že pan Berdár panu Valeskimu členství v představenstvu nabídl a že se dohodli," potvrdil mluvčí Telecomu Vladan Crha. Nabídka podle zdroje přišla poté, co na interním jednání s vedením Eurotelu Berdár ostře kritizoval pomalý rozvoj spolupráce mezi oběma firmami. Telecom v prosinci loňského roku Eurotel zcela ovládl, když dokoupil 49procentní podíl od amerického konsorcia Atlantic West. Podle Crhy by účast Valeskiho v nejvyšším vedení Telecomu měla posílit obchodní spolupráci mateřské a dceřiné firmy a synergie v aktivitách firem. Představenstvo Telecomu má v současnosti pět členů a vedle Berdára jsou jeho členy finanční ředitel Juraj Šedivý, provozní šéf Petr Slováček, obchodní ředitel Roman Stupka a ředitel pro transformaci Michal Heřman. Představenstvo by mělo členství Valeskiho projednat na svém nejbližším zasedání v úterý.
Gold abstract
Generální ředitel Eurotelu Terrence Valeski by mohl zasednout v představenstvu Českého Telecomu. Podle dobře informovaného zdroje mu to nabídl generální ředitel Telecomu Gabriel Berdár a Valeski nabídku přijal. *(Eurotel CEO Terrence Valeski could sit in the Czech Telecom's board of directors. According to a well-informed source Telecom's CEO Gabriel Berdár has offered this and Valeski has accepted the offer.)*
Generated CzeGPT-2 abstract (3 sentences)
• Generální ředitel mobilního operátora Eurotel Terrence Valeski by zasedl v představenstvu Českého Telecomu. *(Eurotel's CEO Terrence Valeski would sit in the board of directors of Czech Telecom.)* • Podle dobře informovaného zdroje mu to nabídl předseda představenstva a generální ředitel Telecomu Gabriel Berdár a Valeski nabídku přijal. *(According to a well-informed source, the chairman of the board and CEO of Telecom Gabriel Berdár offered this and Valeski accepted the offer.)* • "Je pravda, že pan Berdár panu Valeskimu členství v představenstvu nabídl a že se dohodli," potvrdil mluvčí Telecomu Vladan Crha. *("It is true that Mr. Berdár offered Mr. Valeski membership in the board of directors and that they have agreed," Telecom spokesman Vladan Crha confirmed.)*

TABLE 7. Comparison of the inter-annotator agreement measure, Fleiss' κ, among different subsets of the manual evaluation data.

Subset	Special cases	Mapping	Meaning
best	0.78	0.57	0.50
worst	0.62	0.48	0.42
all	0.74	0.58	0.51

summaries are considered good and bad by $\text{ROUGE}_{\text{RAW}}$ and how this is reflected in the actual errors.

After this step, we were left with $4 \times 2 \times 15 = 120$ summaries. We have shuffled the data and created eight random sets, four for each group. These sets were handed to five annotators (all were faculty students and native Czech speakers) for evaluation. The shuffling procedure was performed mainly to keep the sets balanced for the raters to stay alert.

The annotators were provided with an input text (the whole article), the golden summary (original abstract or headline), and the generated summary divided into sentences. Since they did not evaluate the summary's quality and only searched and categorized the errors, the presence of the golden summary was possible and could offer the annotators better orientation in the input text.

The raters went through all the sentences of all generated summaries, and for each, they had two options. Either they found an error and classified it in both the mapping and meaning dimensions, or they picked one of the so-called *Special cases*. The Special cases were *Sentence missing*, *OK*, and *Repetitive (otherwise OK)*. The first value was for the cases when the number of sentences was incorrect, OK was used when no error occurred, and Repetitive was added for situations when the sentence is entirely correct, but it says exactly the same as one of the previous sentences in the generated summary. Such a scenario was not covered in the original methodology, and it has proved helpful in a few cases. The raters were encouraged to explain their error classification using a text box prepared for this purpose within the answer table.

The final error class was decided by a majority of votes. In the case of a draw, we inspected the sentence once again and tried to find an agreement. A generated abstract example with Semantically implausible sentence annotation can be seen in Table 6.

We used the Qualtrics[10] platform to create and distribute the analysis. At the beginning of each set of summaries, we provided our detailed guidelines of the methodology to the annotators.[11]

3) INTERPRETATION

a: FLEISS' κ

To see the consistency of evaluation between the raters, we have computed the inter-annotator agreement (Fleiss' κ) in all three dimensions separately (see Table 7). The values in the range (0.41, 0.60) are generally considered moderately good, and the values in the range (0.60, 0.80) are interpreted as substantial agreement [41]. We can, for example see that annotators have different boundaries for considering a sentence as erroneous (we double-checked these cases, and they were not caused by inattention).

We have also calculated the κ values with each of the raters missing to see if one of them answered differently than the others, but the values were more or less stable.

We can also notice that the agreement is much better for the part of summaries marked as *best*. This might be caused by a higher number of Special cases and undersampled error categories (see Best vs. Worst sets on the facing page).

[10] https://www.qualtrics.com

[11] https://nlp.fi.muni.cz/trac/research/wiki/SummarizationEvaluationManual

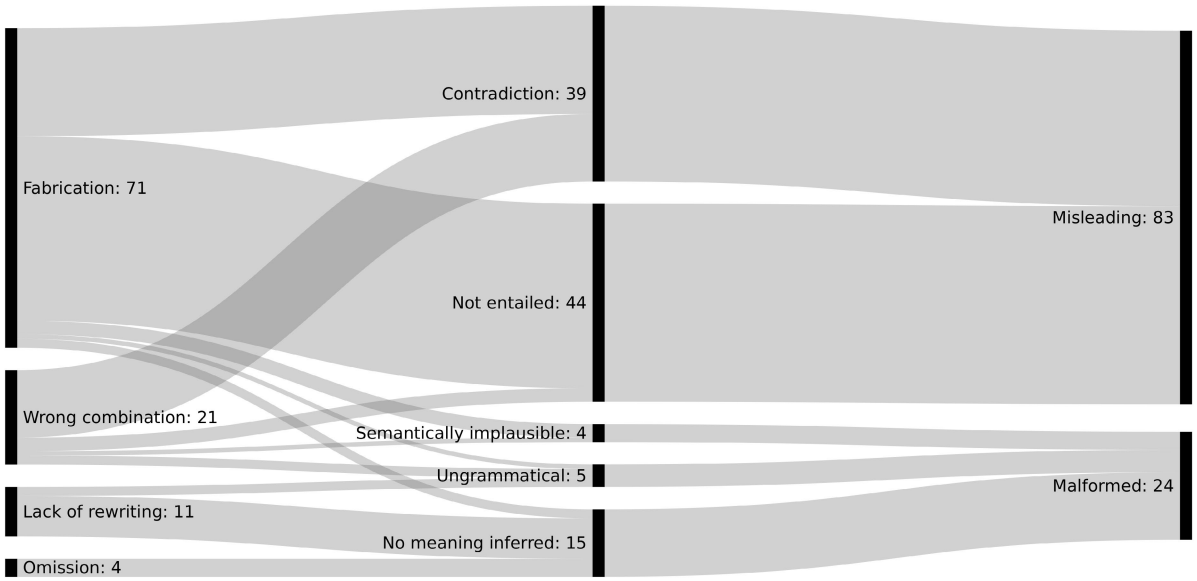

FIGURE 4. Sankey diagram showing the interaction between Mapping and Meaning error dimension.

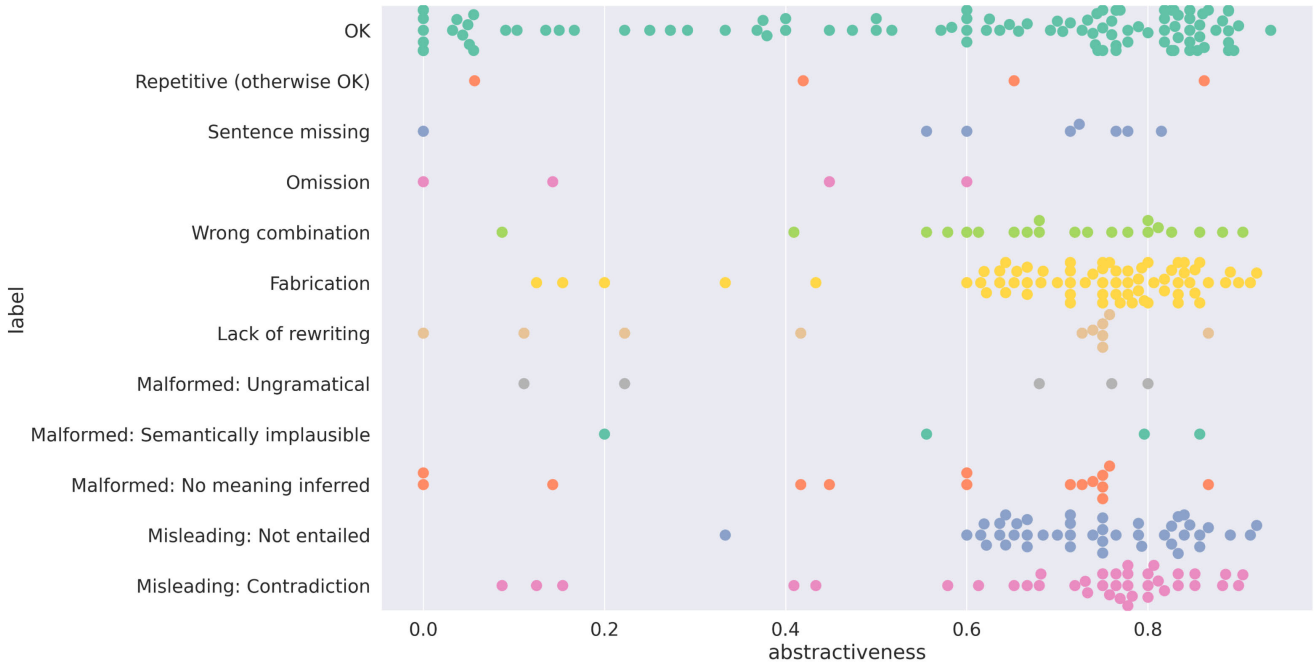

FIGURE 5. Distribution of error types according to the abstractiveness of the generated text.

b: INTERACTION OF MAPPING AND MEANING

Next, we inspected the relationships between the categories falling under Mapping and Meaning (Figure 4). In the Sankey diagrams, we see strong connections between Fabrication → Not entailed and Lack of rewriting → No meaning inferred. These pairs make sense because when you incorporate a word that was not in the input, facts that do not follow the input can be easily introduced. In the latter case, it is often hard to assign a meaning to a sentence when sufficient context is not provided.

c: ABSTRACTIVENESS VS. ERROR

Since with increasing abstractiveness, the amount of rewriting also grows (and it tends to be erroneous), it could be expected that these two variables would correlate. As suggested in [39], we have computed the abstractiveness

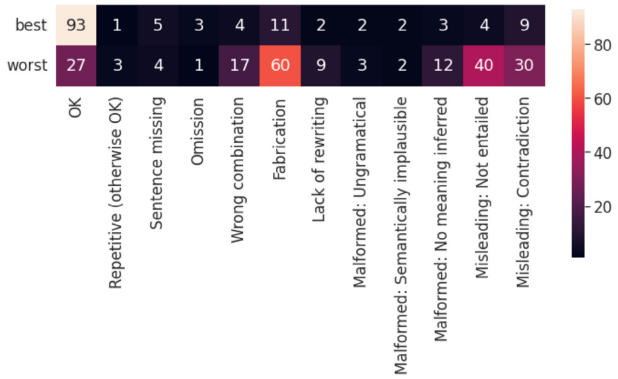

FIGURE 6. Comparison of error types within the *best* and *worst* subgroups.

TABLE 8. Comparison of ratios and accuracies of erroneous and correct sentences among different subsets of the manual evaluation data.

Subset	ratio err : corr	accuracy corr/(err+corr)
best	46 : 139	0.75
headline	32 : 76	0.70
abstract	195 : 271	0.58
worst	181 : 208	0.53
all	203 : 323	0.61

as $\text{ROUGE}_{\text{RAW}}$-L F1 score – the longer common sequence we have, the lower abstractiveness it implies. We can see the trend in Figure 5. Even though the data is biased towards higher abstractiveness, we see a significant predominance of errors on the right side of the plot, while sentences with abstractiveness close to zero are likely to be correct.

d: BEST VS. WORST SETS

Further, we provide a comparison of the errors within the *best* and *worst* subgroups (Figure 6). In the best row, we see a high proportion of correct sentences, accompanied by a pair of small peaks in Fabrication and Contradiction, which seem relevant according to the results from the Sankey diagram. In the "worst" line, the percentage of correct sentences is understandably lower. However, the relationship of Fabrication → Misleading is even more evident.

e: ERROR RATIOS

Finally, we show the ratios of erroneous sentences within different subgroups of the data (see Table 8). The numbers are not directly comparable to any published results because of the unique nature of the subgroups. However, they conclusively confirm the difficulty of correct abstractive summary generation and the insufficiency of the $\text{ROUGE}_{\text{RAW}}$ metric as an evaluation tool.

V. DISCUSSION

As can be seen from the manual error analysis results, the biggest problem with CzeGPT-2 is adding information to the abstract that is not in the input text. This behavior consecutively creates situations where the abstract claims something that we do not find in the original article or even something that contradicts the article's meaning. Unfortunately, this weakness seems to be paradoxically caused by one of the main advantages of the model – the ability to draw knowledge from the pre-training phase. The model is often easily carried away by the information it has encountered in the past and deviates from the sense of the original article. In our view, this issue would not necessarily be solved by a larger model as the "hallucination issue" is present in all current large generative models [42], but rather by an adjusted task-oriented architecture, e.g., a specific focus attention mechanism or a dual attention in encoder-decoder [43].

It also shows that the indisputable advantage of the neural abstractive method is that the model itself decides when to use the extractive and the abstractive approach. The error analysis shows that with increasing extractiveness, the sentences are generally less error-prone, but in some cases, the extractive technique can be too weak. Then, the model needs the ability to improvise. Such a combination is definitely the right step. However, the model must learn when to use which approach.

A. DATASET NOTES

During the manual evaluation, we found out that error analysis is not only helpful in evaluating the summarizer itself. It also pointed out some shortcomings in the dataset, which can affect the results and possibly favor some methods. We often encountered articles that, at the beginning of the section marked as *text* (body of the article), contained a copy of its *abstract*. This makes it an easy job for summarizers that incorporate particular heuristics into their pipeline. An extreme example is the *First* [8] summarizer that only takes the first three sentences of the input text as an abstract.

The dataset also contains structured descriptions of movies or games, where the neural models learn how to answer correctly, but this does not develop a general ability to summarize text.

B. BEST/WORST SELECTION

We would also like to mention the ambiguity of our decision to include the best and worst summaries in the manual evaluation process instead of a random selection suggested by the Lux et al. [39] methodology.

The advantage of our approach is that the choice of the candidate is an exact operation – the set of articles is uniquely determined (if there are several articles with the same ROUGE value, the order depends on the type of sorting algorithm and the dataset order, which is fixed). In contrast, with the original methodology, since the selected set is relatively small, it can easily happen that we do not choose sufficiently representative examples, and the result will be skewed. With different seeds, we can get very different outcomes.

On the other hand, the selection of such two extreme groups may not properly reflect the standard abilities and behavior of the summarizer. At the same time, it is tempting to choose certain types of candidates that, for example, are abundant in the training set. Such types can be the movie or game descriptions mentioned above.

VI. CONCLUSION

In this paper, we have presented the process of training CzeGPT-2, a new autoregressive model by which we expand the ranks of Czech pre-trained transformer models. The model can be broadly utilized and fine-tuned for various downstream tasks, which in some form involve text generation. Here, we have used it as a building block to create a new abstractive summarizer that is compared with the current leading models on the largest Czech summarization dataset.

In standard metrics, CzeGPT-2 surpasses most of the previously published methods besides the significantly larger pretrained mBART large model with state-of-the-art results on both abstract and headline generation tasks. The CzeGPT-2 model is freely available in the standard Hugging Face website for model sharing.[12] With more than 4,000 downloads, the model has already proved useful for various Czech language processing tasks.

Further, we have provided a detailed error analysis of CzeGPT-2 abstractive summarization results that bring us closer to revealing the mechanisms of error generation and their effects on the summary. Although such an analysis is time and human resources-intensive, we want to appeal for tuning a suitable methodology, making it possible to compare abstract summarizers more accurately in the future. The main reason is that the current $\text{ROUGE}_{\text{RAW}}$ metric is not strong enough to reasonably determine the appropriateness or to reveal factual errors in the summaries.

Even though we reached nearly state-of-the-art results, the CzeGPT-2 summarizer can still be improved. Possible further paths can lie in enlarging the model, using task-specific architectures, or augmenting, cleaning, and expanding the dataset. The experience gained on the project is a strong impulse for our future work on this task.

[12]https://huggingface.co/MU-NLPC/CzeGPT-2

REFERENCES

[1] B. Khan, Z. Ali Shah, M. Usman, I. Khan, and B. Niazi, "Exploring the landscape of automatic text summarization: A comprehensive survey," *IEEE Access*, vol. 11, pp. 109819–109840, 2023.

[2] Y. Liu, P. Liu, D. Radev, and G. Neubig, "BRIO: Bringing order to abstractive summarization," in *Proc. 60th Annu. Meeting Assoc. Comput. Linguistics*, vol. 1, Dublin, Ireland, S. Muresan, P. Nakov, and A. Villavicencio, Eds. Stroudsburg, PA, USA: Association for Computational Linguistics, 2022, pp. 2890–2903. [Online]. Available: https://aclanthology.org/2022.acl-long.207/

[3] A. Ghadimi and H. Beigy, "Hybrid multi-document summarization using pre-trained language models," *Expert Syst. Appl.*, vol. 192, Apr. 2022, Art. no. 116292.

[4] A. Alomari, A. S. Al-Shamayleh, N. Idris, A. Q. M. Sabri, I. Alsmadi, and D. Omary, "Warm-starting for improving the novelty of abstractive summarization," *IEEE Access*, vol. 11, pp. 112483–112501, 2023.

[5] H. Y. Koh, J. Ju, M. Liu, and S. Pan, "An empirical survey on long document summarization: Datasets, models, and metrics," *ACM Comput. Surv.*, vol. 55, no. 8, pp. 1–35, Aug. 2023.

[6] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," *OpenAI Blog*, vol. 1, no. 8, p. 9, 2019.

[7] D. Demirci, N. Sahin, M. Sirlancis, and C. Acarturk, "Static malware detection using stacked BiLSTM and GPT-2," *IEEE Access*, vol. 10, pp. 58488–58502, 2022.

[8] M. Straka, N. Mediankin, T. Kocmi, Z. Žabokrtský, V. Hudecek, and J. Hajic, "SumeCzech: Large Czech news-based summarization dataset," in *Proc. 11th Int. Conf. Lang. Resour. Eval.*, Miyazaki, Japan, May 2018, pp. 3488–3495. [Online]. Available: https://aclanthology.org/L18-1551.pdf

[9] Y. Liu, J. Gu, N. Goyal, X. Li, S. Edunov, M. Ghazvininejad, M. Lewis, and L. Zettlemoyer, "Multilingual denoising pre-training for neural machine translation," *Trans. Assoc. Comput. Linguistics*, vol. 8, pp. 726–742, Dec. 2020.

[10] OpenAI et al., "GPT-4 technical report," 2023, *arXiv:2303.08774*.

[11] G. Team et al., "Gemini: A family of highly capable multimodal models," 2023, *arXiv:2312.11805*.

[12] N. Moratanch and S. Chitrakala, "A survey on abstractive text summarization," in *Proc. Int. Conf. Circuit, Power Comput. Technol. (ICCPCT)*, Mar. 2016, pp. 1–7.

[13] I. Sutskever, J. Martens, and G. E. Hinton, "Generating text with recurrent neural networks," in *Proc. 28th Int. Conf. Mach. Learn. (ICML)*, 2011, pp. 1017–1024.

[14] P. Marek, Š. Müller, J. Konrád, P. Lorenc, J. Pichl, and J. Šedivý, "Text summarization of Czech news articles using named entities," *Prague Bull. Math. Linguistics*, vol. 116, no. 1, pp. 5–26, Apr. 2021.

[15] M. Krotil, "Text summarization methods in Czech," M.S. thesis, Faculty Elect. Eng., Dept. Cybern., Czech Tech. Univ., Prague, Czech Republic, 2022. [Online]. Available: https://dspace.cvut.cz/bitstream/handle/10467/101028/F3-BP-2022-Krotil-Marian-CTUWORK-final.pdf

[16] J. Sido, O. Pražák, P. Přibáň, J. Pašek, M. Seják, and M. Konopík, "Czert—Czech BERT-like model for language representation," 2021, *arXiv:2103.13031*.

[17] M. Straka, J. Náplava, J. Straková, and D. Samuel, "RobeCzech: Czech RoBERTa, a monolingual contextualized language representation model," in *Text, Speech, and Dialogue*. Cham, Switzerland: Springer, 2021, pp. 197–209. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-030-83527-9_17#citeas

[18] *Small-E-Czech*, Seznam.cz, Prague, Czechia, 2021. [Online]. Available: https://o.seznam.cz/en/about-us/

[19] A. Kretov, "Attention mechanism in natural language processing," M.S. thesis, Faculty Elect. Eng., Dept. Cybern., Czech Tech. Univ., Prague, Czech Republic, 2020. [Online]. Available: https://dspace.cvut.cz/bitstream/handle/10467/87801/F3-BP-2020-Kretov-Anton-bp_kretov_thesis.pdf

[20] P. Zelina, "Pretraining and evaluation of Czech Albert language model," M.S. thesis, Faculty Inform., Dept. Mach. Learn. Data Process., Fac. Informat. Masaryk Univ., Brno, Czechia, 2020. [Online]. Available: https://is.muni.cz/th/t946b/?lang=en

[21] M. Dalal, A. C. Li, and R. Taori, "Autoregressive models: What are they good for?" 2019, *arXiv:1910.07737*.

[22] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, "Improving language understanding by generative pre-training," OpenAI, San Francisco, CA, USA, Tech. Rep., 2018. [Online]. Available: https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf

[23] T. B. Brown et al., "Language models are few-shot learners," 2020, *arXiv:2005.14165*.

[24] J. Alammar, "The illustrated transformer," Tech. Rep., 2018. [Online]. Available: http://jalammar.github.io/illustrated-transformer/

[25] K. Park, J. Lee, S. Jang, and D. Jung, "An empirical study of tokenization strategies for various Korean NLP tasks," in *Proc. 1st Conf. Asia–Pacific Chapter Assoc. Comput. Linguistics, 10th Int. Joint Conf. Natural Lang. Process.*, Suzhou, China, Dec. 2020, pp. 133–142.

[26] N. Wies, Y. Levine, D. Jannai, and A. Shashua, "Which transformer architecture fits my data? A vocabulary bottleneck in self-attention," in *Proc. Int. Conf. Mach. Learn.*, vol. 139, 2021, pp. 11170–11181.

[27] T. Wolf et al., *Transformers: Perplexity of Fixed-length Models*. Manhattan, NY, USA: Hugging Face, 2020.

[28] C. Huyen, "Evaluation metrics for language modeling," *Gradient*, Oct. 2019. [Online]. Available: https://thegradient.pub/understanding-evaluation-metrics-for-language-models/
[29] C. Wang, M. Li, and A. J. Smola, "Language models with transformers," 2019, *arXiv:1904.09408*.
[30] S. Dudy and S. Bedrick, "Are some words worth more than others?" in *Proc. 1st Workshop Eval. Comparison NLP Syst.*, Nov. 2020, pp. 131–142.
[31] C.-Y. Lin, "ROUGE: A package for automatic evaluation of summaries," in *Proc. ACL Workshop Text Summarization Branches Out*, 2004, pp. 74–81.
[32] J. M. Sanchez-Gomez, M. A. Vega-Rodríguez, and C. J. Pérez, "A multi-objective memetic algorithm for query-oriented text summarization: Medicine texts as a case study," *Expert Syst. Appl.*, vol. 198, Jul. 2022, Art. no. 116769.
[33] P. Guillou, "Faster than training from scratch—Fine-tuning the English GPT-2 in any language with hugging face and fastai v2 (practical case with Portuguese)," Tech. Rep., 2020. [Online]. Available: https://medium.com/@pierre_guillou/faster-than-training-from-scratch-fine-tuning-the-english-gpt-2-in-any-language-with-hugging-f2ec05c98787
[34] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," 2016, *arXiv:1607.04606*.
[35] V. Suchomel, "csTenTen17, a recent Czech web corpus," in *Proc. RASLAN*, 2018, pp. 111–123.
[36] V. Suchomel, "Czech web corpus 2017 (csTenTen17)," LINDAT/CLARIAH-CZ Digit. Library Inst. Formal Appl. Linguistics (ÚFAL), Fac. Math. Phys., Prague, Czechia, Tech. Rep., 4835. [Online]. Available: http://hdl.handle.net/11234/1-4835
[37] L. N. Smith and N. Topin, "Super-convergence: Very fast training of neural networks using large learning rates," 2017, *arXiv:1708.07120*.
[38] L. Xue, N. Constant, A. Roberts, M. Kale, R. Al-Rfou, A. Siddhant, A. Barua, and C. Raffel, "MT5: A massively multilingual pre-trained text-to-text transformer," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Human Lang. Technol.*, 2021, pp. 483–498.
[39] K.-M. Lux, M. Sappelli, and M. Larson, "Truth or error? Towards systematic analysis of factual errors in abstractive summaries," in *Proc. 1st Workshop Eval. Comparison NLP Syst.*, 2020, pp. 1–10.
[40] B. Naismith, P. Mulcaire, and J. Burstein, "Automated evaluation of written discourse coherence using GPT-4," in *Proc. 18th Workshop Innov. Use NLP Building Educ. Appl. (BEA)*, Toronto, ON, Canada, 2023, pp. 394–403.
[41] J. R. Landis and G. G. Koch, "The measurement of observer agreement for categorical data," *Biometrics*, vol. 33, no. 1, p. 159, Mar. 1977.
[42] M. Zhang, O. Press, W. Merrill, A. Liu, and N. A. Smith, "How language model hallucinations can snowball," 2023, *arXiv:2305.13534*.
[43] Z. Ji, N. Lee, R. Frieske, T. Yu, D. Su, Y. Xu, E. Ishii, Y. J. Bang, A. Madotto, and P. Fung, "Survey of hallucination in natural language generation," *ACM Comput. Surv.*, vol. 55, no. 12, pp. 1–38, Dec. 2023.

ADAM HÁJEK is currently pursuing the M.Sc. degree with the Faculty of Informatics, Masaryk University. His research interests include deep learning, large language models, and natural language processing techniques.

ALEŠ HORÁK is currently an Associate Professor of informatics with Masaryk University. His work has been published in, among others, Oxford University Press and Routledge. His research interests include natural language processing, knowledge representation and reasoning, large language models, and corpus linguistics.