Acta Cybernetica — *online–first paper version* — pages 1–15.

Text Cleaning with Transformer Language Models for Hungarian*

Gábor Madarász[ab], András Holl[cd], Noémi Ligeti-Nagy[ae], Zijian Győző Yang[af], and Tamás Váradi[ag]

Abstract

In language technology, clean data is fundamental for training high-quality models, yet large corpora often contain substantial noise due to OCR errors, missing diacritics, and various user-generated inconsistencies. This paper presents a comprehensive text cleaning pipeline tailored for Hungarian, leveraging transformer-based language models optimized for three key tasks: OCR error correction, diacritic restoration, and filtering grammatically incorrect sentences. We introduce huT5, a Hungarian adaptation of the mT5 model, which reduces model parameters and resource demands while maintaining strong performance on Hungarian-specific text cleaning tasks. The huT5 models were fine-tuned on carefully constructed Hungarian corpora for each task and benchmarked against state-of-the-art methods, demonstrating competitive results, particularly in OCR error correction and diacritic restoration. Our pipeline offers an efficient, freely accessible solution to enhance data quality for Hungarian NLP applications, setting a new standard in resource-efficient, language-specific text cleaning.

Keywords: text cleaning, Transformer Language Models, Hungarian NLP, OCR correction, diacritic restoration, huT5 model

1 Introduction

Corpus cleaning is an ongoing challenge in language technology. Clean data is essential for training language models; however, with the large datasets required for deep neural networks, a portion often needs cleaning. Text data can be messy

*This work was supported by the MTA "Tudomány a Magyar Nyelvért Nemzeti Program" (Science for the Hungarian Language National Programme).

[a]HUN-REN Hungarian Research Centre for Linguistics, Budapest, Hungary
[b]E-mail: madarasz.gabor@nytud.hun-ren.hu, ORCID: 0009-0004-8572-3087
[c]Library and Information Centre, Hungarian Academy of Sciences, Budapest, Hungary
[d]E-mail: holl.andras@konyvtar.mta.hu, ORCID: 0000-0002-6873-3425
[e]E-mail: ligeti-nagy.noemi@nytud.hun-ren.hu , ORCID: 0000-0003-0851-7621
[f]E-mail: yang.zijian.gyozo@nytud.hun-ren.hu , ORCID: 0000-0001-9955-860X
[g]E-mail: varadi.tamas@nytud.hun-ren.hu, ORCID: 0000-0001-5765-3908

DOI: 10.14232/actacyb.316365

for various reasons. A prominent research area is the cleaning of OCR-processed texts, which are often prone to errors. Encoding issues, missing diacritics, and grammatical mistakes due to user habits or conventions are also common issues. Cleaning these issues is complex, and no single solution fits all cases. While large language models like ChatGPT hold promise for handling many tasks, they are not perfect and often demand substantial resources, which can be impractical in some scenarios.

To tackle these challenges, we are developing targeted models and applications that can efficiently handle various text cleaning tasks with minimal resources while achieving high accuracy.

Our research prioritizes three main tasks for Hungarian: OCR cleaning, diacritic restoration, and filtering grammatically incorrect texts.

This paper presents a collaborative project between the HUN-REN Hungarian Research Centre for Linguistics (HUN-REN NYTK) and the Library and Information Centre of the Hungarian Academy of Sciences (MTA KIK), aimed at making the contents of the REAL Repository more accessible to researchers and easier to curate and enhance for the MTA KIK.

2 Related Work

The importance of improving texts processed through Optical Character Recognition (OCR) is emphasized by the creation of specialized competitions, such as the post-OCR error correction challenge instituted at the International Conference on Document Analysis and Recognition (ICDAR) since 2017[1]. These competitions have catalyzed progress in OCR error rectification by utilizing cutting-edge methodologies. The method that secured victory in 2019, termed Context-based Character Correction (CCC, [16]), integrated a convolutional neural network with a BERT model for the purpose of error identification, coupled with a bidirectional Long Short-Term Memory (LSTM) encoder-decoder framework incorporating an attention mechanism for error rectification. Other scholars, including [12], adapted CCC by incorporating Named Entity Recognition (NER) for the identification phase and substituting the LSTM-based correction model with a neural machine translation architecture known as OpenNMT. [17] also employed a dual-phase methodology, utilizing a bidirectional LSTM for error identification and a separate LSTM-based model for rectification.

Contemporary advancements have increasingly centered around transformer-based methodologies. [4] addressed OCR correction without employing a specific detection module, instead opting to train a transformer model on a dataset encompassing both accurately transcribed text and OCR-induced erroneous text. They produced synthetic OCR errors through automated techniques, including random error insertion and parallel corpus training, which involved pairing pristine texts with their artificially flawed counterparts. In a similar vein, [9] confronted scenarios involving multiple OCR correction models by devising a statistical approach to

[1] https://sites.google.com/view/icdar2017-postcorrectionocr

rank correction candidates predicated on attributes such as Levenshtein distance and lexicon validation.

[14] conducted an extensive study on types of OCR errors specific to Hungarian. Building on this, [5] used neural language models to correct OCR errors, developing a two-part system with a Context-based Character Correction [12] detection module and an encoder-decoder model fine-tuned to perform the corrections. They also created a manually annotated Gold Standard training corpus for model training.

Numerous studies have explored the examination and cleaning of erroneous Hungarian texts using modern language technology tools. For instance, [3] conducted research on Hungarian corpora to identify the prevalence of non-standard errors, focusing primarily on spoken language and personal subcorpora. Their findings highlighted that the most frequent errors include the omission of punctuation, diacritics, and capitalization. This observation aligns with the fact that Hungarian researchers have undertaken separate studies to address each of these issues. Concerning language correctness, the HuLU benchmark kit includes the Hungarian Corpus of Linguistic Acceptability (HuCOLA) [8], which specifically assesses the grammatical accuracy of Hungarian sentences.

In Hungarian punctuation research, [19] addressed punctuation correction using RNN networks, while [22] approached it with a transformer-based machine translation method.

Diacritic restoration in Hungarian has also been a longstanding area of research. [11] presented a text-to-speech application that handled words without accents. Later, [2] developed an n-gram-based statistical system that restores diacritics without relying on language-specific dictionaries. Several solutions use machine translation techniques to restore accents: [13] employed statistical machine translation, while [6, 7] trained neural machine translation models to solve this problem for Hungarian and other languages.

Despite the progress, most of these models are either not freely accessible or do not align well with our dataset, necessitating the development of custom models to meet our specific needs.

3 Corpora

In our task, we needed to clean a variety of different types of documents in a given library. The Library of the Hungarian Academy of Sciences was established in 1826 and has been serving the members of the Academy and the broader Hungarian research community ever since. The digital collections – in the form of an open access repository – were created in 2008. This repository – named REAL – has diverse holdings, mirroring the printed collection of the library. Its collection includes materials from multiple sources, such as manuscripts, books, and scientific papers, available in formats like printed copies and digital-born versions. This diversity of input channels has resulted in a rich, mixed collection – spanning scanned documents, born-digital files, publishers' PDFs, accepted manuscripts, as well as various handwritten documents and images. For this project, we plan to use a mod-

ern text corpus comprising about 1 billion words. The REAL Repository contains over 250,000 documents, approximately half of which are suitable for our work.

3.1 Training corpora for the cleaning models

For the OCR task, we constructed a custom corpus to train our language model. To ensure accurate error detection, the corpus includes a balanced mix of erroneous and error-free text segment pairs, with a distribution of 66.4% to 33.6%, respectively. For this, we used the corresponding error-free electronic versions of the OCR-processed texts. Table 1 presents the main characteristics of the training and test corpora. The average Character Error Rate (CER) across the entire training dataset is 12.35%, and the Word Error Rate (WER) is 11.74%, measured against the reference data (error-free sentences).

Table 1: Training and test corpora characteristics for the OCR cleaning task

Dataset		Segments	Tokens	Avg Length
Training Set	Source	1,374,665	56,551,620	43.05
	Target	-	47,029,140	35.80
Test Set	Source	6,780	124,401	18.35
	Target	-	115,217	16.99

For diacritic restoration, we used the same corpora as [6] in their research on Hungarian. They chose the online available parallel corpus, Open Subtitles[2], which contains texts written in 62 languages, including Hungarian. It consists of movie subtitles with many shorter, informal sentences in them. Table 2 shows the main characteristics of the training and test corpora. Diacritic characters make up 6.59% of the total characters, and 35.75% of the words contain at least one diacritic.

Table 2: Training and test corpora characteristics for the diacritic restoration task

Dataset	Segments	Tokens	Avg Length
Training Set	28,704,830	177,588,069	6.19
Test Set	3,000	18,635	6.21

For filtering incorrect sentences, we used the HuCOLA corpus from the HuLU benchmark collection [8]. The corpus contains 9076 sentences labelled for their grammaticality. Here we used the training and the validation set, with 7276 and 900 sentences, respectively. Table 3 presents the main characteristics of the training and test corpora, where the distribution of incorrect and correct sentences is 21.6% and 78.4%, respectively.

[2] http://opus.nlpl.eu/OpenSubtitles-v2018.php

Table 3: Characteristics of the HuCOLA training and test corpora

Dataset	Segments	Tokens	Avg Length
Training Set	7,276	50,068	6.88
Test Set	900	6,145	6.75

4 Models and Experiments

Natural language processing has experienced remarkable progress with the advent of neural network technologies, particularly the introduction of the Transformer architecture [20]. The Transformer model has become a foundational milestone in modern language technology, enabling advances in tasks such as translation, summarization, question answering, and supporting our current project. For text-to-text tasks—such as our error correction task—where both the input and output are in textual form, the encoder-decoder architecture used in sequence-to-sequence (seq2seq) models is particularly well-suited. This architecture enables the model to encode the input sequence into a rich contextual representation and then decode it into the desired output format.

4.1 huT5 models

The T5 (Text-To-Text Transfer Transformer) [15], developed by Google Research, introduces a unified framework for natural language processing through transfer learning. It is first pre-trained on a broad, data-rich task to acquire general knowledge, then fine-tuned for specific tasks. T5 treats all NLP tasks—such as translation, classification, or question answering—as text-to-text problems, where both the input and output are in text format. This approach broadens its applicability across diverse NLP tasks. The mT5 [21] model is the multilingual version of T5, trained on the mC4 corpus. mC4 is a multilingual variant of the C4 (Colossal Clean Crawled Corpus) dataset, containing texts in 101 different languages, including Hungarian.

Based on related work [5, 6], we identified the encoder-decoder architecture [20] as the most effective for our needs. However, no pretrained encoder-decoder language models specifically for Hungarian were available. Consequently, we developed a custom model by adapting the mT5 for Hungarian. We followed guidance on single-language adaptation from a multilingual T5 model[3], using methods to prune redundant embeddings and reduce parameter counts with minimal quality loss. Additionally, we optimized vocabulary size based on [1]. For the Hungarian adaptation, we used a vocabulary trained on a Hungarian dataset to filter out non Hungarian-specific tokens from the original vocabulary. This process resulted in

[3] https://towardsdatascience.com/how-to-adapt-a-multilingual-t5-model-for-a-single-language-b9f94f3d9c90

two Hungarian-specific mT5 models (huT5) in both base and large versions, which will be made available on our Huggingface site[4].

Once the huT5 models were created, we fine-tuned them for OCR cleaning, diacritic restoration, and incorrect sentence filtering.

4.2 Text cleaning pipeline

Using our fine-tuned, task-specific models, we can build a text cleaning pipeline to address our task. Figure 1 shows the architecture of our text cleaning pipeline. In our task, the primary errors originate from OCR; therefore, our first module is an OCR cleaner. The second major source of errors is missing or incorrect diacritics, so our second module is a diacritic restoration model. After these two modules complete the cleaning, if errors still remain, we use an erroneous sentence detector to mark or filter out the incorrect sentences.

Figure 1: Architecture of text cleaning pipeline

In the next section, we will provide a detailed introduction to each of these modules.

4.3 Modules and models

For the OCR cleaning task, we initially tested the model from [5], but it did not perform consistently across all text types in our dataset. Similarly, previous models for diacritic restoration [6] were not freely available, prompting us to train custom models for these tasks.

Therefore, for these tasks, we fine-tuned custom, task-specific models based on our huT5 models. The large models were trained using two NVIDIA A100 GPUs (80GB each), while the base models were trained on four NVIDIA GeForce GTX 1080 GPUs (11GB each). We utilized the PyTorch Seq2SeqTrainer from the Transformers library[5] for fine-tuning. For comparison, we also trained the original mT5 models on these tasks.

The training hyperparameters for the OCR cleaning models are as follows: learning rate = 5e-5; global batch size = 256; epoch = 1; sequence length = 256.

The training hyperparameters for the diacritic restoration models are as follows: learning rate = 5e-5; global batch size = 512; epoch = 1; sequence length = 256.

For both OCR cleaning and diacritic restoration tasks, we tailored the batch sizes to the available GPU and used gradient accumulation to achieve the global

[4] https://huggingface.co/NYTK
[5] https://github.com/huggingface/transformers/tree/main/examples/pytorch

batch size. Due to the larger corpus size in diacritic restoration, we selected a larger global batch size for this task.

The training hyperparameters for the HuCOLA models are as follows: learning rate = 5e-6; global batch size = 32; epoch = 10; sequence length = 128. In the HuCOLA task, we trained for 10 epochs to ensure reliable comparisons, selecting the best-performing checkpoint. In the case of the HuCOLA experiment, we did not use gradient accumulation. For better comparison, the global batch size was kept the same as in the experiment conducted by [24].

In the final text cleaning pipeline, the best-performing models constitute the OCR cleaning, diacritic restoration, and erroneous filtering modules.

To evaluate our huT5 models comprehensively, we also fine-tuned both huT5 and mT5 models on the Hungarian Corpus of Linguistic Acceptability (HuRTE) and Hungarian version of the Stanford Sentiment Treebank (HuRTE) benchmarks [8].

5 Results

Our first objective was to evaluate the huT5 models. Table 4 provides a comparison between our huT5 and mT5 models. A key result is the significant reduction in both the number of parameters (base: ∼42%; large: ∼68%) and model size (base: ∼42%; large: ∼67%) following the conversion. For quality evaluation, we used the HuCOLA, HuSST, and HuRTE benchmarks, along with accuracy as the evaluation metric. The primary focus here is on the performance difference between the two model types rather than on outperforming existing Hungarian models.

The results indicate that in all cases, the huT5 models perform similarly or better than the mT5 models, despite having fewer parameters.

Consistent with findings from [23, 25], the encoder-decoder architecture tends to be more suitable for sequence-to-sequence tasks than for classification. This is reflected in our experiments with Hungarian benchmarks, where the models showed lower performance on classification tasks.

Table 4: Comparison of huT5 and mT5 on Hungarian benchmarks

Model	Parameters	Size	HuCOLA	HuRTE	HuSST
huT5 base	244 million	977 MB	**80.98**	**55.00**	58.37
mT5 base	580 million	2.33 GB	**80.98**	52.66	57.94
huT5 large	820 million	3.28 GB	80.88	54.00	**58.79**
mT5 large	1.2 billion	4.92 GB	80.88	53.50	58.02

Given that encoder-decoder models generally perform better on text generation tasks, we expect this model to be well-suited for our OCR cleaning and diacritic restoration tasks. In these experiments, we compared our models with Hungarian

state-of-the-art (SOTA) models in this field, with benchmark results available in [5] and [6]. The results can be seen in Table 5.

For evaluation, we used ROUGE-L [18] (where higher scores indicate better quality ↑), Word Error Rate (WER, with lower values indicating better performance ↓), and Character Error Rate (CER, with lower values indicating better performance ↓) [10] for OCR cleaning. For diacritic restoration, we assessed model performance using precision and recall metrics.

As you can see, our huT5 large model achieved the best performance in OCR cleaning, outperforming the SOTA model. In the diacritic restoration task, none of our models surpassed the SOTA models, but we achieved competitive results and will publish our models. Notably, our huT5 models, with fewer parameters, outperformed or achieved competitive performance compared to the mT5 models.

Table 5: Comparison of OCR cleaning and diacritic restoration results for huT5 and mT5 models

Model	OCR Cleaning			Diacritic Restoration	
	ROUGE-L ↑	WER ↓	CER ↓	Precision ↑	Recall ↑
SOTA model	93.44	17.38	8.72	99.38	99.28
huT5 base	95.12	11.10	7.05	97.64	97.75
mT5 base	95.26	10.61	6.57	97.57	97.49
huT5 large	**95.66**	**10.01**	**6.46**	97.95	98.18
mT5 large	95.28	11.56	9.33	98.61	98.61

The third module is an erroneous sentence detector. After the previous two modules have corrected the text, any remaining errors can, depending on the task, be marked or filtered out by an error detector. For this task, the HuCOLA benchmark is the most suitable. As shown in Table 4, our model achieved only about 80% accuracy. In comparison, the models trained by [24] performed better, with the HuBERT and PULI models achieving approximately 90% and 91% accuracy, respectively. Consequently, in our final pipeline, we use the fine-tuned PULI Bert-Large model instead of our fine-tuned huT5 or mT5.

We also performed error analysis on the OCR and diacritic restoration models, which achieved the best performance.

In Table 6, the error analysis of the diacritic restoration model is presented. The typical errors are similar to those reported by [6], with only the ratios differing. In our case, we observed a higher proportion of real errors.

There are two main categories included in the correct output: equivalent forms and correct replaceable outputs. The equivalent form (e.g., *hova-hová* 'where', *tied-tiéd* 'yours') refers to cases where both forms of the given word are usable, with no difference in meaning. The replaceable outputs refer to cases where the given words have different meanings, but both are correct either in the given context or without additional context. In the case of real errors, approximately 30% of the errors stem from proper nouns, as the model is unable to correctly restore names.

Table 6: Error analysis of the diacritic restoration model

Error type	**Ratio**	**Examples** (reference (ref) - prediction (pred))
Correct output	**37.5%**	
Equivalent form		*hova - hová* ('where'), *tied - tiéd* ('yours')
Replaceable output		ref: *Fogjátok meg!* ('Catch him/her!') pred: *Fogjatok meg!* ('Catch me!') ref: *Az InStyle magazinnal.* ('With InStyle magazine'.) pred: *Az InStyle magazinnál.* ('At InStyle magazine.')
Wrong reference		ref: *Két kijárat, egy elől* ('from'), *egy hátul.* pred: *Két kijárat, egy elöl* ('in front'), *egy hátul.*
Real errors	**62.5%**	
Proper noun		*Liúról - Liuról*, *Ramával - Rámával*
Wrongly replaced		*még - meg*, *teli - téli*

In Table 7, the error analysis of the OCR cleaning model is presented. Notably, 21% of the reported errors are actually false positives, where the reference text was incorrect, and the model provided the correct output. The remaining 79% of the errors represent real mistakes. These errors can be classified into three main categories: insertion, deletion, and replacement:

- Insertion: This occurs when the model adds an incorrect character to a word. Common cases involve the addition of extra punctuation marks or entirely incorrect letters. Additionally, the model may erroneously insert a space within a word, leading to segmentation errors. Finally, there are cases where the model inserts words that did not originally exist in the text.

- Deletion: In this category, the model removes a character that should have been retained. Similar to the insertion category, these deletions can involve punctuation marks, letters, spaces, or entire words. Deletion of spaces often results in the unintended merging of two words, while the deletion of whole words leads to a loss of information.

- Replacement: This category encompasses two primary types of errors: the substitution of an incorrect punctuation mark or letter. A notable subtype of replacement errors involves the mishandling of diacritic marks, where the model incorrectly replaces accented characters. Additionally, there are cases where the model replaces an entire word with a completely unrelated one, resulting in semantic deviations.

This analysis highlights specific patterns of error generation, providing insights for targeted model improvements.

Table 7: Error analysis of the OCR cleaning

Error type	**Ratio**	**Examples** (reference (ref) - prediction (pred))
Correct output	**21.38%**	*pcdig - pedig* *minder.kinek - mindenkinek* *szinház- színház*
Real errors	**78,62%**	
Insertion	23,83%	*írva - írva*: (punctuation mark) *darab - dakrab* (letter) *mennybéli - menny béli* (space) *németalföldi - német 376 alföldi* (words)
Deletion	24,44%	*halastó, - halastó* (punctuation mark) *vesszővel - vesszvel* (letter) *már most - mármost* (space) *Hozott Isten!* - [] (words)
Replacement	30,35%	*bort, - bort;* (punctuation mark) *előbb - elébb* (letter) *színes - szines* (diacritic) *Mi - Ali* (word)

6 Challenges and limitations

Adapting mT5 to huT5 presented several challenges, particularly in addressing the unique linguistic features of Hungarian, such as complex morphology and extensive diacritic use. These characteristics required substantial pre-processing to enhance model robustness, especially for tasks like diacritic restoration and grammatical error detection. Additionally, as Hungarian is relatively low-resource, the limited availability of high-quality annotated datasets constrained training, which may affect model performance on domain-specific or informal text corpora.

Another challenge was the computational demand required for model adaptation and fine-tuning. Despite efforts to reduce parameter count, the large huT5 version remains resource-intensive, potentially limiting its accessibility to groups without advanced hardware. Additionally, adapting huT5 to specific tasks required careful tuning to avoid overfitting due to dataset limitations.

Lastly, current evaluation metrics like WER, CER, and ROUGE-L offer a partial view of improvements in text quality and readability. Developing refined evaluation metrics tailored to Hungarian text cleaning could help better capture the model's impact.

7 Conclusion

In this study, we addressed the essential task of text cleaning for Hungarian using custom transformer-based models. The results of our OCR cleaning, diacritic restoration, and incorrect sentence filtering tasks highlight the effectiveness and adaptability of our huT5 models, showing both improved performance and resource efficiency over existing models. By adapting the mT5 model specifically for Hungarian, we achieved substantial reductions in parameter counts and model size, with a remarkable 42% reduction for the base model and 68% for the large model. This optimization not only maintained but also, in many cases, improved the model's performance on Hungarian benchmarks, particularly in sequence-to-sequence tasks.

Our results confirm that the huT5 models are well-suited for a range of text cleaning tasks. Compared to the original mT5, the huT5 models consistently achieved better scores on Hungarian OCR cleaning and diacritic restoration, as shown by lower WER and CER values and higher ROUGE-L scores. Additionally, the high precision and recall in diacritic restoration and the solid performance across all metrics make the huT5 models a strong candidate for state-of-the-art solutions in Hungarian text cleaning.

This work also provides a valuable, freely accessible alternative for Hungarian-language tasks, meeting gaps left by existing models. The encoder-decoder architecture used in our approach effectively addresses both sequence-to-sequence and error detection needs, presenting a refined tool for improving data quality in Hungarian NLP applications.

As a next step, we would like to experiment with merging the OCR cleaner model and the diacritic restoration model. We believe that a larger model can effectively solve both problems within a single architecture.

Similar to the work of Laki and Yang [6], our diacritic restoration experiment can be extended to a multilingual setting.

Future work may extend these results by testing the huT5 models' adaptability to additional languages or dialects. Nonetheless, this study establishes a new standard for Hungarian text cleaning, showing that transformer-based approaches, when tailored to the specific language requirements, can achieve both high accuracy and efficiency.

References

[1] Abdaoui, A., Pradel, C., and Sigel, G. Load what you need: Smaller versions of mutililingual BERT. In Moosavi, N. S., Fan, A., Shwartz, V., Glavaš, G., Joty, S., Wang, A., and Wolf, T., editors, *Proceedings of SustaiNLP: Workshop on Simple and Efficient Natural Language Processing*, pages 119–123. Association for Computational Linguistics, 2020. DOI: 10.18653/v1/2020.sustainlp-1.16.

[2] Ács, J. and Halmai, J. Hunaccent: Small footprint diacritic restoration for social media. In *Proceedings of the Tenth International Conference on Language*

Resources and Evaluation, pages 3526–3529, Portoroz, Slovenia, 2016. URL: https://hlt.bme.hu/media/pdf/acs_halmi_2016.pdf.

[3] Dömötör, A. and Yang, Z. Gy. Így írtok ti: nem sztenderd szövegek hibatípusainak detektálása gépi tanulásos módszerrel [This is how you write: detecting error types in non-standard texts using machine learning]. In Vincze, V., editor, *XIV. Magyar Számítógépes Nyelvészeti Konferencia*, pages 305–316, Szeged, Hungary, 2018. Szegedi Tudományegyetem, Informatikai Tanszékcsoport. URL: https://rgai.inf.u-szeged.hu/sites/rgai.sed.hu/files/teljesB5.pdf.

[4] Duong, Q., Hämäläinen, M., and Hengchen, S. An unsupervised method for OCR post-correction and spelling normalisation for Finnish. In *Proceedings of the 23rd Nordic Conference on Computational Linguistics*, pages 240–248, Reykjavik, Iceland (Online), 2021. Linköping University Electronic Press, Sweden. URL: https://aclanthology.org/2021.nodalida-main.24/.

[5] Laki, L. J., Kőrös, A., Ligeti-Nagy, N., Nyéki, B., Vadász, N., Yang, Z. Gy., and Váradi, T. OCR-hibák javítása neurális technológiák segítségével [Correcting OCR errors using neural technologies]. In *XVIII. Magyar Számítógépes Nyelvészeti Konferencia*, pages 417–430, Szeged, Hungary, 2022. Szegedi Tudományegyetem, Informatikai Intézet. URL: https://rgai.inf.u-szeged.hu/sites/rgai.inf.u-szeged.hu/files/mszny2022.pdf.

[6] Laki, L. J. and Yang, Z. Gy. Automatic diacritic restoration with transformer model based neural machine translation for East-Central European languages. In *Proceedings of the 11th International Conference on Applied Informatics*, pages 190–202, Eger, Hungary, 2020. URL: http://ceur-ws.org/Vol-2650/.

[7] Laki, L. J. and Yang, Z. Gy. Automatikus ékezetvisszaállítás transzformer modellen alapuló neurális gépi fordítással [Automatic accent restoration using neural machine translation based on transformer model]. In Berend, G., Gosztolya, G., and Vincze, V., editors, *XVI. Magyar Számítógépes Nyelvészeti Konferencia*, pages 181–190, Szeged, Hungary, 2020. Szegedi Tudományegyetem, Informatikai Tanszékcsoport. URL: https://rgai.inf.u-szeged.hu/sites/rgai.inf.u-szeged.hu/files/mszny2020_0.pdf.

[8] Ligeti-Nagy, N., Ferenczi, G., Héja, E., Laki, L. J., Vadász, N., Yang, Z. G., and Váradi, T. HuLU: Hungarian language understanding benchmark kit. In Calzolari, N., Kan, M.-Y., Hoste, V., Lenci, A., Sakti, S., and Xue, N., editors, *Proceedings of the Joint International Conference on Computational Linguistics, Language Resources and Evaluation*, pages 8360–8371, Torino, Italia, 2024. ELRA and ICCL. URL: https://aclanthology.org/2024.lrec-main.733.

[9] Mei, J., Islam, A., Wu, Y., Moh'd, A., and Milios, E. E. Statistical learning for OCR text correction. *CoRR*, abs/1611.06950, 2016. DOI: 10.48550/arXiv.1611.06950.

[10] Morris, A., Maier, V., and Green, P. From WER and RIL to MER and WIL: improved evaluation measures for connected speech recognition. In *Interspeech 2004*, pages 2765–2768, 2004. DOI: 10.21437/Interspeech.2004-668.

[11] Németh, G., Zainkó, Cs., Fekete, L., Olaszy, G., Endrédi, G., Olaszi, P., Kiss, G., and Kis, P. The design, implementation, and operation of a Hungarian e-mail reader. *International Journal of Speech Technology*, 3(3):217–236, 2000. DOI: 10.1023/A:1026567216832.

[12] Nguyen, T. T. H., Jatowt, A., Nguyen, N.-V., Coustaty, M., and Doucet, A. Neural machine translation with BERT for post-OCR error detection and correction. In *Proceedings of the ACM/IEEE Joint Conference on Digital Libraries*, page 333–336, New York, NY, USA, 2020. Association for Computing Machinery. DOI: 10.1145/3383583.3398605.

[13] Novák, A. and Siklósi, B. Automatic diacritics restoration for Hungarian. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 2286–2291, Lisbon, Portugal, 2015. Association for Computational Linguistics. DOI: 10.18653/v1/D15-1275.

[14] Pethő, G., Sass, B., Simon, L., and Lipp, V. OCR-hibák kvantitatív elemzése több szövegváltozat összehasonlításával [Quantitative analysis of OCR errors through the comparison of multiple text versions]. In *XX. Magyar Számítógépes Nyelvészeti Konferencia*, pages 17–29, Szeged, Hungary, 2024. Szegedi Tudományegyetem, Informatikai Intézet. URL: https://rgai.inf.u-szeged.hu/sites/rgai.inf.u-szeged.hu/files/mszny2024%20%281%29.pdf.

[15] Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(1):5485–5551, 2020. URL: https://dl.acm.org/doi/10.5555/3455716.3455856.

[16] Rigaud, C., Doucet, A., Coustaty, M., and Moreux, J.-P. ICDAR 2019 competition on post-OCR text correction. In *Proceedings of the International Conference on Document Analysis and Recognition*, pages 1588–1593, 2019. DOI: 10.1109/ICDAR.2019.00255.

[17] Schaefer, R. and Neudecker, C. A two-step approach for automatic OCR post-correction. In *Proceedings of the 4th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*, pages 52–57. International Committee on Computational Linguistics, 2020. URL: https://aclanthology.org/2020.latechclfl-1.6/.

[18] See, A., Liu, P. J., and Manning, C. D. Get to the point: Summarization with pointer-generator networks. In Barzilay, R. and Kan, M.-Y., editors, *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada,

2017. Association for Computational Linguistics. DOI: 10.18653/v1/P17-1099.

[19] Tündik, M. A., Tarján, B., and Szaszák, G. Televíziós feliratok írásjeleinek visszaállítása rekurrens neurális hálózatokkal [Restoration of punctuation marks in television subtitles using Recurrent Neural Networks]. In Vincze, V., editor, *XIV. Magyar Számítógépes Nyelvészeti Konferencia*, pages 183–195, Szeged, Hungary, 2018. Szegedi Tudományegyetem, Informatikai Tanszékcsoport. URL: https://rgai.inf.u-szeged.hu/sites/rgai.sed.hu/files/teljesB5.pdf.

[20] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., and Polosukhin, I. Attention is all you need. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc., 2017. URL: http://papers.nips.cc/paper/7181-attention-is-all-you-need.pdf.

[21] Xue, L., Constant, N., Roberts, A., Kale, M., Al-Rfou, R., Siddhant, A., Barua, A., and Raffel, C. mT5: A massively multilingual pre-trained text-to-text transformer. In Toutanova, K., Rumshisky, A., Zettlemoyer, L., Hakkani-Tur, D., Beltagy, I., Bethard, S., Cotterell, R., Chakraborty, T., and Zhou, Y., editors, *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498. Association for Computational Linguistics, 2021. DOI: 10.18653/v1/2021.naacl-main.41.

[22] Yang, Z. Gy. Automatikus írásjelek visszaállítása és Nagybetűsítés statikus korpuszon, transzformer modellen alapuló neurális gépi fordítással [Automatic punctuation restoration and capitalization on a static corpus using transformer-based neural machine translation]. In Berend, G., Gosztolya, G., and Vincze, V., editors, *XVII. Magyar Számítógépes Nyelvészeti Konferencia*, pages 225–232, Szeged, Hungary, 2021. Szegedi Tudományegyetem, Informatikai Tanszékcsoport. URL: https://rgai.inf.u-szeged.hu/sites/rgai.inf.u-szeged.hu/files/mszny2021.pdf.

[23] Yang, Z. Gy. BARTerezzünk! – Messze, messze, messze a világtól, – BART kísérleti modellek magyar nyelvre [Let's BART! – Far, far, far away from the world, – BART experimental models for Hungarian]. In *XVIII. Magyar Számítógépes Nyelvészeti Konferencia*, pages 15–29, Szeged, Hungary, 2022. Szegedi Tudományegyetem, Informatikai Intézet. URL: https://rgai.inf.u-szeged.hu/sites/rgai.inf.u-szeged.hu/files/mszny2022.pdf.

[24] Yang, Z. Gy., Dodé, R., Ferenczi, G., Héja, E., Jelencsik-Mátyus, K., Kőrös, A., Laki, L. J., Ligeti-Nagy, N., Vadász, N., and Váradi, T. Jönnek a nagyok! BERT-Large, GPT-2 és GPT-3 nyelvmodellek magyar nyelvre

[The giants are coming! BERT-Large, GPT-2, and GPT-3 language models for Hungarian]. In *XIX. Magyar Számítógépes Nyelvészeti Konferencia*, pages 247–262, Szeged, Hungary, 2023. Szegedi Tudományegyetem, Informatikai Intézet. URL: https://rgai.inf.u-szeged.hu/sites/rgai.inf.u-szeged.hu/files/mszny2023_0.pdf.

[25] Yang, Z. Gy. and Váradi, T. Training language models with low resources: RoBERTa, BART and ELECTRA experimental models for Hungarian. In *Proceedings of 12th IEEE International Conference on Cognitive Infocommunications (CogInfoCom 2021)*, pages 279–285. IEEE, 2021. URL: https://m2.mtmt.hu/api/publication/32651934.