Heidelberg-Boston @ SIGTYP 2024 Shared Task: Enhancing Low-Resource Language Analysis With Character-Aware Hierarchical Transformers

Frederick Riemenschneider*
Dept. of Computational Linguistics
Heidelberg University, Germany
riemenschneider@cl.uni-heidelberg.de

Kevin Krahn*
Dept. of Computer Science
Sattler College, USA
kevin.krahn24@sattler.edu

Abstract

Historical languages present unique challenges to the NLP community, with one prominent hurdle being the limited resources available in their closed corpora. This work describes our submission to the constrained subtask of the SIGTYP 2024 shared task, focusing on PoS tagging, morphological tagging, and lemmatization for 13 historical languages. For PoS and morphological tagging we adapt a hierarchical tokenization method from Sun et al. (2023) and combine it with the advantages of the DeBERTa-V3 architecture, enabling our models to efficiently learn from every character in the training data. We also demonstrate the effectiveness of character-level T5 models on the lemmatization task. Pre-trained from scratch with limited data, our models achieved first place in the constrained subtask, nearly reaching the performance levels of the unconstrained task's winner. Our code is available at https://github.com/bowphs/SIGTYP-2024-hierarchical-transformers.

1 Introduction

Unlike modern languages, historical languages come with a notable challenge: their corpora are closed, meaning they cannot grow any further. This situation often puts researchers of historical languages in a low-resource setting, requiring tailored strategies to handle language processing and analysis effectively (Johnson et al., 2021).

In this paper, we focus on identifying the most efficient methods for extracting information from small corpora. In such a scenario, the main hurdle is not computational capacity, but learning to extract the maximal amount of information from our existing data.

To evaluate this, the SIGTYP 2024 shared task offers a targeted platform centering on the evaluation of embeddings and systems for historical languages. This task provides a systematic testbed for researchers, allowing us to assess our methodologies in a controlled evaluation setting for historical language processing.

For the constrained subtask, participants received annotated datasets for 13 historical languages sourced from Universal Dependencies (Zeman et al., 2023), along with data for Old Hungarian that adheres to similar annotation standards (Simon, 2014; HAS Research Institute for Linguistics, 2018). These languages represent four distinct language families and employ six different scripts, which ensures a high level of diversity. The rules imposed in this subtask strictly forbid the use of pre-trained models and limit training exclusively to the data of the specified language. This restriction not only ensures full comparability of the applied methods, it also inhibits any cross-lingual transfer effects.

We demonstrate that, even in these resource-limited settings, it is feasible to achieve high performance using monolingual models. Our models are exclusively pre-trained on very small corpora, leveraging recent advances in pre-training language models. Our submission was recognized as the winner in the constrained task. Notably, it also delivered competitive results in comparison to the submissions in the unconstrained task, where the use of additional data was permitted. This highlights the strength of our approach, even within a more restricted data environment.

2 Pre-trained Language Models for Ancient and Historical Languages

Much of the previous work on Pre-trained Language Models (PLMs) for ancient and historical languages has focused on cross-lingual transfer learning techniques (Krahn et al., 2023; Singh et al., 2021; Yamshchikov et al., 2022; Yousef et al., 2022) or languages with relatively large corpora compared to most historical languages, such as An-

*Equal contribution.

Proceedings of the 6th Workshop on Research in Computational Linguistic Typology and Multilingual NLP (SIGTYP 2024), pages 131–141
March 22, 2024

Language:	chu	cop	fro	got	grc	hbo	isl	lat	latm	lzh	ohu	orv	san
Vocab Size:	196	82	106	87	242	94	150	188	111	5714	166	222	62

Table 1: Character vocabulary sizes (including special tokens). See Appendix C for language identifiers.

cient Greek and Latin (Riemenschneider and Frank, 2023; Bamman and Burns, 2020). In this work, we are interested in maximizing performance in more resource-limited environments while training exclusively on monolingual data.

2.1 Representing Words and Characters

Low-resource historical languages present several challenges for subword tokenizers which are typically used by PLMs. Given that our downstream tasks require predictions at the world level, it is important that the model learns good word representations in training. At the same time, it is important to obtain good character representations because characters carry important morphological information. In small-scale training corpora, subword tokenizers are ineffective at capturing information at both the word and character levels, as shown in prior work (Clark et al., 2022; Kann et al., 2018). As a result, it is difficult for a model to learn meaningful representations for rare tokens, which can be completely opaque to the model with respect to the characters they contain.

Adopting a character-based tokenizer would solve many of these problems, but as a downside would result in a much higher number of input tokens. Critically, the computational requirements of self-attention grow quadratically with sequence length, making training and inference time prohibitive or requiring truncated input sequences.

For these reasons, we adopt a solution for our encoder-only models that combines the advantages of word- and character-level representations. We base our architecture on the Hierarchical Pretrained Language Model (HLM) architecture recently proposed by Sun et al. (2023), which solves many of our problems. HLM is a hierarchical two-level model which uses a shallow intra-word transformer encoder to learn word representations from characters and a deep inter-word encoder that attends to the entire word sequence. As a result, (1) it gives direct access to characters without requiring long sequence lengths, (2) it preserves explicit word boundaries, and (3) it allows for an open vocabulary.

For the intra-word encoder, we use a sequence length of 16 which is long enough to cover the vast majority of words in our training data. While Sun et al. (2023) truncate words that exceed the maximum sequence length of the intra-word encoder, we instead split them into multiple subwords to avoid any loss of information. For the inter-word encoder we use a maximum sequence length of 512. Because the intra-word encoder is limited to characters within the same word and the inter-word encoder operates on word sequences, this approach is computationally more efficient than a vanilla character model, and even approaches the performance of subword-based models (Sun et al., 2023).

The input to the intra-word encoder is produced by encoding each word into a sequence of character tokens, with a special `[WORD_CLS]` token inserted at the beginning of each word. The contextualized `[WORD_CLS]` embeddings from the intra-word encoder are then used as the word representations for the inter-word encoder.

We create a character tokenizer for each language using a character vocabulary consisting of all the unique characters found in the training data for that language. Any unseen characters encountered in the validation or test data are replaced with a special `[UNK]` token. Table 1 shows the vocabulary sizes for each language, including special tokens. The character vocabularies are typically quite small, with the notable exception of Classical Chinese (lzh), where most of the tokens in the training data are single characters. We experimented with several decomposition methods, inspired by the work of Si et al. (2023) on sub-character tokenization for Chinese. However, we were unable to improve performance on our downstream tasks, so we opted to use the same character tokenization method for all languages.

2.2 Hierarchical Encoder-only Models

To conduct PoS and morphological tagging, we rely on an encoder that generates the necessary word embeddings for classification. Our encoder models build on a modified implementation of DeBERTa-V3 (He et al., 2023), combining the advantages of HLM with the DeBERTa architecture. The intra-

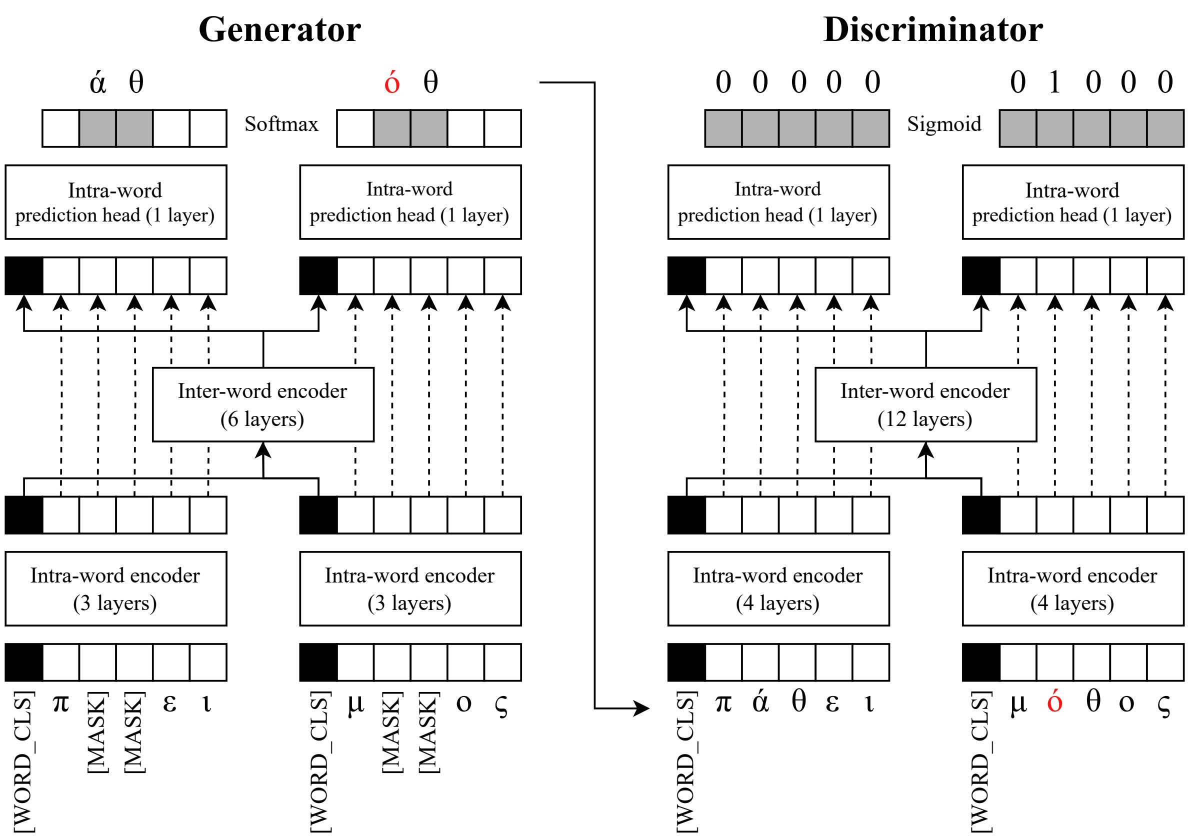

Figure 1: HLM-DeBERTa architecture with RTD pre-training. Input text is “πάθει μάθος”.

and inter-word modules are implemented as two separate DeBERTa encoders, utilizing disentangled attention (He et al., 2021) and relative position encoding.

Replaced Token Detection. For the pre-training task we use replaced token detection (RTD), originally proposed by Clark et al. (2020). RTD uses a generator model to generate corrupted input sequences and a discriminator to distinguish between the original and corrupted tokens. After training, the generator is discarded and the discriminator is fine-tuned for downstream tasks. In our experiments, when applying RTD pre-training, we achieve slightly better performance on our downstream tasks compared to masked language modeling (MLM) as the pre-training task. Following previous work (He et al., 2023; Clark et al., 2020), we use a generator with roughly half the model parameters compared to the discriminator. We train a monolingual model for each language for 30 epochs. Further pre-training does not improve performance on downstream tasks.

We utilize DeBERTa-V3’s gradient-disentangled embedding sharing (GDES), which allows the embedding gradients from the generator to flow directly to the discriminator, but not vice versa. This results in more stable training compared to the vanilla embedding sharing (ES) used by ELECTRA (Clark et al., 2020), which allows the gradients to flow in both directions.

Masking Strategy. We use character-level masking to allow for open-vocabulary language modeling. The character token sequence is restored by concatenating the character representations from the intra-word module with the word representations from the inter-word module, replacing the initial `[WORD_CLS]` with the contextualized representation. We follow the original HLM approach for the language modeling prediction head: an additional single-layer intra-word transformer module followed by a simple feed-forward network. A softmax layer is used for the generator’s output distribution and a sigmoid layer is used for the discriminator. The relative position embedding matrix is shared between the initial intra-word encoder and the intra-word language modeling head. Figure 1 shows an overview of our architecture for RTD pre-training.

We compare the following masking strategies:

- Whole-word masking: mask the characters in 15% of the words (original HLM approach),
- Character masking: randomly mask 15% of the characters,
- Character n-gram masking: mask random spans of 1-4 characters until 15% of the characters are masked.

Through experimentation we found that character n-gram masking performed best for our downstream tasks, by a small margin. Random character masking performed similarly to whole-word-masking. We hypothesize that it is too difficult for the model to learn to predict whole words from the small training corpora. Conversely, random character masking is too easy, as MLM pre-training accuracy reaches high levels very quickly.

2.3 Character-level Encoder-decoder Models

While encoder-only models are very effective for classification tasks, lemmatization is most naturally treated as a sequence-to-sequence problem, where the inflected form is "translated" to its lemma. We therefore choose to train an encoder-decoder model that handles sequence-to-sequence tasks naturally. Specifically, we train a T5 model for each language (Raffel et al., 2020) using the nanoT5 library (Nawrot, 2023) and the t5-v1_1-base configuration. In lemmatization, our aim is to prioritize the characters within a word, rather than focusing on a detailed understanding of contextualized words (see Section 3.3 for our approach). Moreover, extending a hierarchical structure to (encoder-)decoder models like T5 is not straightforward. Therefore, we employ character tokenization in the T5 models for lemmatization.

3 Using our PLMs for Downstream Tasks

Many systems focusing on Universal Dependencies, often introduced in shared tasks, utilize cross-lingual transfer and multi-task learning. For instance, UDPipe (Straka et al., 2019), which employs multilingual BERT, is fine-tuned on specific treebanks for PoS tagging, morphological tagging, lemmatization, and dependency parsing. UDify (Kondratyuk and Straka, 2019) learns these tasks for 75 languages in one model.

Given that in our setting cross-lingual transfer is excluded, we investigate multi-task learning as a remaining option to leverage additional training signals for resource-poor languages.

3.1 Morphological Tagging

Following Riemenschneider and Frank (2023), we treat morphological tagging as a multi-task-classification problem, where every token is processed through k classification heads, corresponding to each possible morphological feature in a dataset. Whenever a feature is missing in a token, the model is trained to predict a class indicating the feature's absence.

To represent a token, the HLM architecture yields two kinds of embeddings: those derived from the intra-word encoder, informed by a word's characters but not by other sentence words, and those that are contextualized by surrounding tokens. In line with Sun et al. (2023) as well as earlier work (Clark et al., 2022; Plank et al., 2016), we concatenate these embeddings to create a unified final word representation.

We use a simple feed-forward network followed by a softmax function on top of the last hidden state of this word representation. The final loss is computed as:

$$\mathcal{L}_{\text{morph}} = \frac{1}{k} \sum_{m=0}^{k-1} \mathcal{L}_m$$

where k is the number of morphological features.

We further extended the multi-task framework to include additional related tasks, hypothesizing that obtaining training signals from auxiliary tasks could improve the model's capabilities, particularly under our low-resource conditions. To this end, we incorporated tasks such as dependency parsing and PoS tagging. Contrary to our expectations, this approach led to slower convergence and did not provide any performance benefits, occasionally even producing marginally inferior results. We discuss these findings in Section 5.

3.2 PoS Tagging

Analogous to our approach in morphological tagging, we represent each token by concatenating its intra- and inter-word embeddings, followed by a classification head. However, in contrast to morphological tagging, we notice slight improvements when the model is also tasked with predicting morphological features. Thus, we determine the loss as $\mathcal{L}_{\text{UPoS}} + \mathcal{L}_{\text{morph}}$, disregarding the morphological tagging predictions during inference.

3.3 Lemmatization

As outlined in Section 2.3, lemmatization is most naturally treated as a sequence-to-sequence problem, where the form to be lemmatized is transduced into its lemma, which is why we propose using a T5 model for this task. Ideally, our model should receive the word to be lemmatized in its original context, while marking the word to be lemmatized, similar to the approach used by Riemenschneider and Frank (2023). For instance, given the input sequence ξύνοιδα [SEP] ἐμαυτῷ [SEP] οὐδὲν ἐπισταμένῳ, the model would be expected to predict the lemma of ἐμαυτῷ, which is ἐμαυτοῦ. This approach would enable us to train the model in an end-to-end fashion, allowing it to autonomously learn the relevant information directly from the word within its contextual surroundings.

However, this training method is prohibitively expensive, requiring repeated passes through the model, once for each token in the sentence. Moreover, we noted that the models exhibited exceptionally slow convergence. Allowing the model to predict lemmata for all words in a sentence in a single forward pass mitigates the computational challenges, as it requires only one pass per sentence per epoch. Yet, this strategy still encounters problems with very slow, and at times nonexistent, convergence, while also introducing new challenges for the model, particularly in assigning exactly one lemma to each token accurately.

Therefore, we adopt a pipeline approach, following Wróbel and Nowak (2022), by providing the model with the inflected form and its corresponding UPoS tag. For training purposes, we use the gold UPoS tag, whereas for inference we rely on the UPoS tag as predicted by our HLM-DeBERTa model. We predict lemmata using beam search with a beam width of 20, restricting the maximum sequence length to 30.

4 Results

Our results are computed using the SIGTYP 2024 official evaluation script.[1] The script computes PoS tagging scores as the unweighted average of the accuracy and the F_1 score. For morphological tagging, it computes the averaged accuracy across each token, with deductions for any feature categories predicted by the model but absent in the label. The lemmatization scores are the unweighted average of the accuracy@1 and the accuracy@3.

We report our results in Table 2 and provide dataset statistics in Appendix C. In **PoS** and **morphological tagging**, our system emerges as the winner of the constrained task. Its performance is consistently almost on-par with that of the unconstrained task winner, being only 0.69 percentage points lower on average. A notable outlier is seen in Old French (fro) PoS tagging, where our system falls short by 3 percentage points. This performance difference might be linked to the small size of the Old French corpus in the treebank, although our model generally shows strong performance in learning from small datasets, as demonstrated by its robust performance in other datasets of similar size, such as Ancient Hebrew (hbo), Gothic (got), and Vedic Sanskrit (san).

Results in **lemmatization** display greater diversity, likely due to the differing architectures in participants' approaches. Our model achieves 99.18% in Classical Chinese (lzh), a language where distinct lemmata do not really exist, usually turning the task into mere form replication. This score, though precise, is somewhat lower than the near-perfect range of 99.81 to 99.96% achieved by the other methods in the shared task.

5 Negative Results

Multi-task Learning. We hypothesized that a model simultaneously doing PoS tagging, morphological tagging and dependency parsing could benefit from the training signals of related tasks.[2] However, this approach did not significantly improve morphological analysis and resulted in longer training times due to slower convergence. On the other hand, jointly performing morphological and PoS tagging in a multi-task learning setup yielded minor improvements in PoS tagging. We believe that including PoS information offers little extra insight to the model for morphological tagging and simultaneously pressures it to form representations apt for PoS tagging. Conversely, enriching the coarser PoS tagging task with morphological labels provides the model with useful additional insights. Furthermore, our dependency parsing technique differs from the more direct classification approach used in PoS and morphological tagging, potentially leading to instabilities during training.

[1] https://github.com/sigtyp/ST2024/blob/main/scoring_program_constrained.zip.

[2] For dependency parsing, we adopt the head selection method as described by Zhang et al. (2017).

	Language:	chu	cop	fro	got	grc	hbo	isl	lat	latm	lzh	ohu	orv	san
Morphological Tagging														
Constrained	Ours	**96.04**	**98.60**	**97.87**	**95.32**	**97.46**	**97.46**	**95.29**	**95.17**	**98.68**	**95.52**	**96.30**	**95.00**	**91.58**
	Team 21a	94.06	80.47	94.08	93.96	96.50	71.20	94.79	93.31	97.98	85.98	94.64	92.16	90.00
	Baseline	85.07	47.41	28.27	18.95	25.10	42.78	35.83	18.17	30.94	43.58	23.20	25.55	08.34
Unconstrained	UDParse	**96.49**	**98.88**	**98.33**	**96.23**	**97.78**	**97.05**	**95.92**	**96.66**	**98.83**	**96.24**	**96.62**	**95.16**	**92.60**
	TartuNLP	67.14	74.86	98.01	92.40	97.33	95.14	95.53	95.91	**98.83**	88.75	75.62	80.00	86.33
PoS Tagging														
Constrained	Ours	**96.57**	**96.92**	**93.10**	**95.41**	**96.39**	**96.68**	**96.08**	**95.54**	**98.43**	**92.92**	**95.98**	**94.46**	**89.71**
	Team 21a	94.62	42.65	85.14	93.48	93.49	27.26	93.85	92.43	94.41	81.79	94.42	91.23	87.32
	Baseline	93.36	94.98	91.57	93.73	90.33	94.07	94.00	92.39	97.22	90.91	93.59	90.33	89.37
Unconstrained	UDParse	**97.00**	**97.33**	**96.01**	**96.47**	**96.49**	**97.84**	**96.88**	**96.83**	**98.79**	**93.76**	**96.71**	**94.99**	**90.02**
	TartuNLP	66.35	60.99	94.51	92.72	95.72	94.15	96.67	95.86	**98.79**	83.28	75.14	75.67	83.83
Lemmatization														
Constrained	Ours	**94.49**	95.07	**92.63**	**93.31**	**94.08**	**97.29**	**96.63**	**96.00**	**98.46**	99.18	85.92	**90.09**	**84.59**
	Team 21a	79.59	46.32	83.32	90.79	88.30	61.75	94.58	92.35	97.22	**99.84**	69.97	78.44	83.21
	Baseline	89.60	**95.74**	91.93	91.95	91.06	95.28	93.78	92.08	97.03	98.81	**89.43**	84.44	84.24
Unconstrained	UDParse	59.56	74.78	92.47	92.81	**94.02**	96.85	**97.96**	**96.74**	**98.91**	**99.96**	63.43	68.55	88.10
	TartuNLP	**92.70**	**98.28**	**95.11**	**95.41**	93.39	**98.15**	97.23	**96.99**	98.69	99.91	**86.91**	**89.23**	**91.48**

Table 2: Results on *SIGTYP 2024 Shared Task on Word Embedding Evaluation for Ancient and Historical Languages*. We mark the winner of each subtask in **bold** and underline the overall winner. See Appendix C for language identifiers.

Tall Models. Xue et al. (2023) found that transformers with a narrower and deeper architecture might surpass the performance of similarly sized models in masked language modeling tasks. Inspired by this finding, we experimented with doubling the number of layers to 24 while reducing the hidden size from 768 to 512 and the number of attention heads from 12 to 8. However, although this adjustment seemed to yield a marginal improvement in pre-training with MLM, it did not result in any performance changes when training with RTD.

6 Conclusion

We present our approach for the SIGTYP 2024 shared task on historical language analysis. Our method employs a hierarchical transformer that first focuses on a word's characters, applying self-attention to generate initial word embeddings. These embeddings are then further developed by integrating the contextual information from surrounding words. We pre-train HLM-DeBERTa-V3 and T5 models with small datasets of historical texts. The character-based methodology of our architecture yielded promising results, effectively leveraging the available data. Contrary to our expectations, the implementation of multi-task learning had only a negligible effect on enhancing our models' performance.

Acknowledgements

We thank Anette Frank for her helpful suggestions and her constructive feedback on our paper. We are deeply grateful to Fabian Strobel for his support and the valuable pointers he provided.

References

David Bamman and Patrick J. Burns. 2020. Latin BERT: A contextual language model for classical philology.

Jonathan H. Clark, Dan Garrette, Iulia Turc, and John Wieting. 2022. Canine: Pre-training an efficient tokenization-free encoder for language representation. *Transactions of the Association for Computational Linguistics*, 10:73–91.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: Pre-training text encoders as discriminators rather than generators. In *ICLR*.

HAS Research Institute for Linguistics. 2018. Old Hungarian Codices.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2023. DeBERTav3: Improving DeBERTa using ELECTRA-style pre-training with gradient-disentangled embedding sharing. In *The Eleventh International Conference on Learning Representations*.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. DeBERTa: Decoding-enhanced BERT with disentangled attention. In *International Conference on Learning Representations*.

Kyle P. Johnson, Patrick J. Burns, John Stewart, Todd Cook, Clément Besnier, and William J. B. Mattingly. 2021. The Classical Language Toolkit: An NLP framework for pre-modern languages. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations*, pages 20–29, Online. Association for Computational Linguistics.

Katharina Kann, Johannes Bjerva, Isabelle Augenstein, Barbara Plank, and Anders Søgaard. 2018. Character-level supervision for low-resource POS tagging. In *Proceedings of the Workshop on Deep Learning Approaches for Low-Resource NLP*, pages 1–11, Melbourne. Association for Computational Linguistics.

Dan Kondratyuk and Milan Straka. 2019. 75 languages, 1 model: Parsing Universal Dependencies universally. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2779–2795, Hong Kong, China. Association for Computational Linguistics.

Kevin Krahn, Derrick Tate, and Andrew C. Lamicela. 2023. Sentence embedding models for Ancient Greek using multilingual knowledge distillation. In *Proceedings of the Ancient Language Processing Workshop*, pages 13–22, Varna, Bulgaria. INCOMA Ltd., Shoumen, Bulgaria.

Piotr Nawrot. 2023. nanoT5: Fast & simple pre-training and fine-tuning of t5 models with limited resources. In *Proceedings of the 3rd Workshop for Natural Language Processing Open Source Software (NLP-OSS 2023)*, pages 95–101, Singapore. Association for Computational Linguistics.

Barbara Plank, Anders Søgaard, and Yoav Goldberg. 2016. Multilingual part-of-speech tagging with bidirectional long short-term memory models and auxiliary loss. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 412–418, Berlin, Germany. Association for Computational Linguistics.

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

Frederick Riemenschneider and Anette Frank. 2023. Exploring large language models for classical philology. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15181–15199, Toronto, Canada. Association for Computational Linguistics.

Chenglei Si, Zhengyan Zhang, Yingfa Chen, Fanchao Qi, Xiaozhi Wang, Zhiyuan Liu, Yasheng Wang, Qun Liu, and Maosong Sun. 2023. Sub-character tokenization for Chinese pretrained language models. *Transactions of the Association for Computational Linguistics*, 11:469–487.

Eszter Simon. 2014. Corpus Building from Old Hungarian Codices. In Katalin É. Kiss, editor, *The Evolution of Functional Left Peripheries in Hungarian Syntax*. Oxford University Press, Oxford.

Pranaydeep Singh, Gorik Rutten, and Els Lefever. 2021. A pilot study for BERT language modelling and morphological analysis for ancient and medieval Greek. In *Proceedings of the 5th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*, pages 128–137, Punta Cana, Dominican Republic (online). Association for Computational Linguistics.

Milan Straka, Jana Straková, and Jan Hajič. 2019. Evaluating contextualized embeddings on 54 languages in POS tagging, lemmatization and dependency parsing. *arXiv preprint arXiv:1908.07448*.

Li Sun, Florian Luisier, Kayhan Batmanghelich, Dinei Florencio, and Cha Zhang. 2023. From characters to words: Hierarchical pre-trained language model for open-vocabulary language understanding. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3605–3620, Toronto, Canada. Association for Computational Linguistics.

Krzysztof Wróbel and Krzysztof Nowak. 2022. Transformer-based part-of-speech tagging and lemmatization for Latin. In *Proceedings of the*

Second Workshop on Language Technologies for Historical and Ancient Languages, pages 193–197, Marseille, France. European Language Resources Association.

Fuzhao Xue, Jianghai Chen, Aixin Sun, Xiaozhe Ren, Zangwei Zheng, Xiaoxin He, Yongming Chen, Xin Jiang, and Yang You. 2023. A study on transformer configuration and training objective. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org.

Ivan Yamshchikov, Alexey Tikhonov, Yorgos Pantis, Charlotte Schubert, and Jürgen Jost. 2022. BERT in plutarch's shadows. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6071–6080, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Tariq Yousef, Chiara Palladino, Farnoosh Shamsian, Anise d'Orange Ferreira, and Michel Ferreira dos Reis. 2022. An automatic model and gold standard for translation alignment of Ancient Greek. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 5894–5905, Marseille, France. European Language Resources Association.

Daniel Zeman, Joakim Nivre, Mitchell Abrams, Elia Ackermann, Noëmi Aepli, Hamid Aghaei, Željko Agić, Amir Ahmadi, Lars Ahrenberg, Chika Kennedy Ajede, Salih Furkan Akkurt, Gabrielė Aleksandravičiūtė, Ika Alfina, Avner Algom, Khalid Alnajjar, Chiara Alzetta, Erik Andersen, Lene Antonsen, Tatsuya Aoyama, Katya Aplonova, Angelina Aquino, Carolina Aragon, Glyd Aranes, Maria Jesus Aranzabe, Bilge Nas Arıcan, Ĥórunn Arnardóttir, Gashaw Arutie, Jessica Naraiswari Arwidarasti, Masayuki Asahara, Katla Ásgeirsdóttir, Deniz Baran Aslan, Cengiz Asmazoğlu, Luma Ateyah, Furkan Atmaca, Mohammed Attia, Aitziber Atutxa, Liesbeth Augustinus, Mariana Avelãs, Elena Badmaeva, Keerthana Balasubramani, Miguel Ballesteros, Esha Banerjee, Sebastian Bank, Verginica Barbu Mititelu, Starkaður Barkarson, Rodolfo Basile, Victoria Basmov, Colin Batchelor, John Bauer, Seyyit Talha Bedir, Shabnam Behzad, Kepa Bengoetxea, İbrahim Benli, Yifat Ben Moshe, Gözde Berk, Riyaz Ahmad Bhat, Erica Biagetti, Eckhard Bick, Agnė Bielinskienė, Kristín Bjarnadóttir, Rogier Blokland, Victoria Bobicev, Loïc Boizou, Emanuel Borges Völker, Carl Börstell, Cristina Bosco, Gosse Bouma, Sam Bowman, Adriane Boyd, Anouck Braggaar, António Branco, Kristina Brokaitė, Aljoscha Burchardt, Marisa Campos, Marie Candito, Bernard Caron, Gauthier Caron, Catarina Carvalheiro, Rita Carvalho, Lauren Cassidy, Maria Clara Castro, Sérgio Castro, Tatiana Cavalcanti, Gülşen Cebiroğlu Eryiğit, Flavio Massimiliano Cecchini, Giuseppe G. A. Celano, Slavomír Čéplö, Neslihan Cesur, Savas Cetin, Özlem Çetinoğlu, Fabricio Chalub, Liyanage Chamila, Shweta Chauhan, Ethan Chi, Taishi Chika, Yongseok Cho, Jinho Choi, Jayeol Chun, Juyeon Chung, Alessandra T. Cignarella, Silvie Cinková, Aurélie Collomb, Çağrı Çöltekin, Miriam Connor, Daniela Corbetta, Francisco Costa, Marine Courtin, Mihaela Cristescu, Ingerid Løyning Dale, Philemon Daniel, Elizabeth Davidson, Leonel Figueiredo de Alencar, Mathieu Dehouck, Martina de Laurentiis, Marie-Catherine de Marneffe, Valeria de Paiva, Mehmet Oguz Derin, Elvis de Souza, Arantza Diaz de Ilarraza, Carly Dickerson, Arawinda Dinakaramani, Elisa Di Nuovo, Bamba Dione, Peter Dirix, Kaja Dobrovoljc, Adrian Doyle, Timothy Dozat, Kira Droganova, Puneet Dwivedi, Christian Ebert, Hanne Eckhoff, Masaki Eguchi, Sandra Eiche, Marhaba Eli, Ali Elkahky, Binyam Ephrem, Olga Erina, Tomaž Erjavec, Farah Essaidi, Aline Etienne, Wograine Evelyn, Sidney Facundes, Richárd Farkas, Federica Favero, Jannatul Ferdaousi, Marília Fernanda, Hector Fernandez Alcalde, Amal Fethi, Jennifer Foster, Cláudia Freitas, Kazunori Fujita, Katarína Gajdošová, Daniel Galbraith, Federica Gamba, Marcos Garcia, Moa Gärdenfors, Fabrício Ferraz Gerardi, Kim Gerdes, Luke Gessler, Filip Ginter, Gustavo Godoy, Iakes Goenaga, Koldo Gojenola, Memduh Gökırmak, Yoav Goldberg, Xavier Gómez Guinovart, Berta González Saavedra, Bernadeta Griciūtė, Matias Grioni, Loïc Grobol, Normunds Grūzītis, Bruno Guillaume, Céline Guillot-Barbance, Tunga Güngör, Nizar Habash, Hinrik Hafsteinsson, Jan Hajič, Jan Hajič jr., Mika Hämäläinen, Linh Hà Mỹ, Na-Rae Han, Muhammad Yudistira Hanifmuti, Takahiro Harada, Sam Hardwick, Kim Harris, Dag Haug, Johannes Heinecke, Oliver Hellwig, Felix Hennig, Barbora Hladká, Jaroslava Hlaváčová, Florinel Hociung, Petter Hohle, Marivel Huerta Mendez, Jena Hwang, Takumi Ikeda, Anton Karl Ingason, Radu Ion, Elena Irimia, Ọlájídé Ishola, Artan Islamaj, Kaoru Ito, Siratun Jannat, Tomáš Jelínek, Apoorva Jha, Katharine Jiang, Anders Johannsen, Hildur Jónsdóttir, Fredrik Jørgensen, Markus Juutinen, Hüner Kaşıkara, Nadezhda Kabaeva, Sylvain Kahane, Hiroshi Kanayama, Jenna Kanerva, Neslihan Kara, Ritván Karahóǧa, Andre Kåsen, Tolga Kayadelen, Sarveswaran Kengatharaiyer, Václava Kettnerová, Jesse Kirchner, Elena Klementieva, Elena Klyachko, Arne Köhn, Abdullatif Köksal, Kamil Kopacewicz, Timo Korkiakangas, Mehmet Köse, Alexey Koshevoy, Natalia Kotsyba, Jolanta Kovalevskaitė, Simon Krek, Parameswari Krishnamurthy, Sandra Kübler, Adrian Kuqi, Oğuzhan Kuyrukçu, Aslı Kuzgun, Sookyoung Kwak, Kris Kyle, Veronika Laippala, Lorenzo Lambertino, Tatiana Lando, Septina Dian Larasati, Alexei Lavrentiev, John Lee, Phương Lê Hồng, Alessandro Lenci, Saran Lertpradit, Herman Leung, Maria Levina, Lauren Levine, Cheuk Ying Li, Josie Li, Keying Li, Yixuan Li, Yuan Li, KyungTae Lim, Bruna Lima Padovani, Yi-Ju Jessica Lin, Krister Lindén, Yang Janet Liu, Nikola Ljubešić, Olga Loginova, Stefano Lusito, Andry Luthfi, Mikko Luukko, Olga Lyashevskaya, Teresa Lynn, Vivien Macketanz, Menel Mahamdi, Jean Maillard, Ilya Makarchuk, Aibek Makazhanov, Michael Mandl, Christopher Manning, Ruli Manurung, Büşra Marşan, Cătălina Mărănduc, David Mareček, Katrin Marheinecke, Stella Markantonatou, Héctor Martínez Alonso, Lorena

Martín Rodríguez, André Martins, Cláudia Martins, Jan Mašek, Hiroshi Matsuda, Yuji Matsumoto, Alessandro Mazzei, Ryan McDonald, Sarah McGuinness, Gustavo Mendonça, Tatiana Merzhevich, Niko Miekka, Aaron Miller, Karina Mischenkova, Anna Missilä, Cătălin Mititelu, Maria Mitrofan, Yusuke Miyao, AmirHossein Mojiri Foroushani, Judit Molnár, Amirsaeid Moloodi, Simonetta Montemagni, Amir More, Laura Moreno Romero, Giovanni Moretti, Shinsuke Mori, Tomohiko Morioka, Shigeki Moro, Bjartur Mortensen, Bohdan Moskalevskyi, Kadri Muischnek, Robert Munro, Yugo Murawaki, Kaili Müürisep, Pinkey Nainwani, Mariam Nakhlé, Juan Ignacio Navarro Horñiacek, Anna Nedoluzhko, Gunta Nešpore-Bērzkalne, Manuela Nevaci, Lương Nguyễn Thị, Huyền Nguyễn Thị Minh, Yoshihiro Nikaido, Vitaly Nikolaev, Rattima Nitisaroj, Alireza Nourian, Hanna Nurmi, Stina Ojala, Atul Kr. Ojha, Hulda Óladóttir, Adédayọ Olúòkun, Mai Omura, Emeka Onwuegbuzia, Noam Ordan, Petya Osenova, Robert Östling, Lilja Øvrelid, Şaziye Betül Özateş, Merve Özçelik, Arzucan Özgür, Balkız Öztürk Başaran, Teresa Paccosi, Alessio Palmero Aprosio, Anastasia Panova, Hyunji Hayley Park, Niko Partanen, Elena Pascual, Marco Passarotti, Agnieszka Patejuk, Guilherme Paulino-Passos, Giulia Pedonese, Angelika Peljak-Łapińska, Siyao Peng, Siyao Logan Peng, Rita Pereira, Sílvia Pereira, Cenel-Augusto Perez, Natalia Perkova, Guy Perrier, Slav Petrov, Daria Petrova, Andrea Peverelli, Jason Phelan, Jussi Piitulainen, Yuval Pinter, Clara Pinto, Tommi A Pirinen, Emily Pitler, Magdalena Plamada, Barbara Plank, Thierry Poibeau, Larisa Ponomareva, Martin Popel, Lauma Pretkalniņa, Sophie Prévost, Prokopis Prokopidis, Adam Przepiórkowski, Robert Pugh, Tiina Puolakainen, Sampo Pyysalo, Peng Qi, Andreia Querido, Andriela Rääbis, Alexandre Rademaker, Mizanur Rahoman, Taraka Rama, Loganathan Ramasamy, Joana Ramos, Fam Rashel, Mohammad Sadegh Rasooli, Vinit Ravishankar, Livy Real, Petru Rebeja, Siva Reddy, Mathilde Regnault, Georg Rehm, Arij Riabi, Ivan Riabov, Michael Rießler, Erika Rimkutė, Larissa Rinaldi, Laura Rituma, Putri Rizqiyah, Luisa Rocha, Eiríkur Rögnvaldsson, Ivan Roksandic, Mykhailo Romanenko, Rudolf Rosa, Valentin Roșca, Davide Rovati, Ben Rozonoyer, Olga Rudina, Jack Rueter, Kristján Rúnarsson, Shoval Sadde, Pegah Safari, Aleksi Sahala, Shadi Saleh, Alessio Salomoni, Tanja Samardžić, Stephanie Samson, Manuela Sanguinetti, Ezgi Sanıyar, Dage Särg, Marta Sartor, Mitsuya Sasaki, Baiba Saulīte, Yanin Sawanakunanon, Shefali Saxena, Kevin Scannell, Salvatore Scarlata, Nathan Schneider, Sebastian Schuster, Lane Schwartz, Djamé Seddah, Wolfgang Seeker, Mojgan Seraji, Syeda Shahzadi, Mo Shen, Atsuko Shimada, Hiroyuki Shirasu, Yana Shishkina, Muh Shohibussirri, Maria Shvedova, Janine Siewert, Einar Freyr Sigurðsson, João Silva, Aline Silveira, Natalia Silveira, Sara Silveira, Maria Simi, Radu Simionescu, Katalin Simkó, Mária Šimková, Haukur Barri Símonarson, Kiril Simov, Dmitri Sitchinava, Ted Sither, Maria Skachedubova, Aaron Smith, Isabela Soares-Bastos, Per Erik Solberg, Barbara Sonnenhauser, Shafi Sourov, Rachele Sprugnoli, Vivian Stamou, Steinh͡ór Steingrímsson, Antonio Stella, Abishek Stephen, Milan Straka, Emmett Strickland, Jana Strnadová, Alane Suhr, Yogi Lesmana Sulestio, Umut Sulubacak, Shingo Suzuki, Daniel Swanson, Zsolt Szántó, Chihiro Taguchi, Dima Taji, Fabio Tamburini, Mary Ann C. Tan, Takaaki Tanaka, Dipta Tanaya, Mirko Tavoni, Samson Tella, Isabelle Tellier, Marinella Testori, Guillaume Thomas, Sara Tonelli, Liisi Torga, Marsida Toska, Trond Trosterud, Anna Trukhina, Reut Tsarfaty, Utku Türk, Francis Tyers, Sveinbjörn H͡órðarson, Vilhjálmur H͡orsteinsson, Sumire Uematsu, Roman Untilov, Zdeňka Urešová, Larraitz Uria, Hans Uszkoreit, Andrius Utka, Elena Vagnoni, Sowmya Vajjala, Socrates Vak, Rob van der Goot, Martine Vanhove, Daniel van Niekerk, Gertjan van Noord, Viktor Varga, Uliana Vedenina, Giulia Venturi, Veronika Vincze, Natalia Vlasova, Aya Wakasa, Joel C. Wallenberg, Lars Wallin, Abigail Walsh, Jonathan North Washington, Maximilan Wendt, Paul Widmer, Shira Wigderson, Sri Hartati Wijono, Seyi Williams, Mats Wirén, Christian Wittern, Tsegay Woldemariam, Tak-sum Wong, Alina Wróblewska, Mary Yako, Kayo Yamashita, Naoki Yamazaki, Chunxiao Yan, Koichi Yasuoka, Marat M. Yavrumyan, Arife Betül Yenice, Olcay Taner Yıldız, Zhuoran Yu, Arlisa Yuliawati, Zdeněk Žabokrtský, Shorouq Zahra, Amir Zeldes, He Zhou, Hanzhi Zhu, Yilun Zhu, Anna Zhuravleva, and Rayan Ziane. 2023. Universal dependencies 2.12. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Xingxing Zhang, Jianpeng Cheng, and Mirella Lapata. 2017. Dependency parsing as head selection. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 665–676, Valencia, Spain. Association for Computational Linguistics.

A Pre-Training Details

Parameter	Generator	Discriminator
Activation	GELU	GELU
Hidden Dropout	0.1	0.1
Initializer Range	0.02	0.02
Intra-word encoder		
Layers	3	4
Hidden Size	768	768
Intermediate Size	1536	1536
Attention Heads	12	12
Inter-word encoder		
Layers	6	12
Hidden Size	768	768
Intermediate Size	3072	3072
Attention Heads	12	12

Table 3: HLM-DeBERTa hyperparameters.

Parameter	Value
Optimizer	Adam
Weight Decay	0.01
Batch Size	16
Learning Rate	1e-5
Learning Rate Scheduler	constant
Epochs	30
Warmup Proportion	0.1
Mask Percentage	15%
Max Sequence Length (words)	512
Max Word Length (chars)	16

Table 4: HLM-DeBERTa pre-training hyperparameters.

Parameter	Value
Optimizer	AdamWScale*
Weight Decay	0.0
Batch Size	16
Learning Rate	1e-5
Learning Rate Scheduler	cosine
Epochs	100
Warmup Steps	1000
Mask Percentage	15%
Max Sequence Length	512
Mean Noise Span Length	3

Table 6: T5 pre-training hyperparameters.

* We use the customized AdamW implementation of nanoT5 (Nawrot, 2023) that is augmented by RMS scaling.

Parameter	Encoder	Decoder
Activation	GEGLU	GEGLU
Hidden Dropout	0.0	0.0
Layers	12	12
Hidden Size	768	768
Intermediate Size	2048	2048
Attention Heads	12	12

Table 5: T5 hyperparameters.

B Fine-tuning Details

Parameter	Value
Optimizer	AdamW
Weight Decay	0.01
Batch Size	16
Learning Rate	2e-5
Learning Rate Scheduler	linear
Early Stopping Patience	10

Table 7: HLM-DeBERTa fine-tuning hyperparameters.

Parameter	Value
Optimizer	AdamW
Weight Decay	0.01
Batch Size	16
Learning Rate	1e-3
Learning Rate Scheduler	linear
Early Stopping Patience	10

Table 8: T5 fine-tuning hyperparameters.

C Dataset Statistics

Language	Code	Family	Script	Train Tok.	Valid Tok.	Test Tok.	Train Sent.	Valid Sent.	Test Sent.
Ancient Greek	grc	Indo-European	Greek	334 043	41 905	41 046	24 800	3100	3101
Ancient Hebrew	hbo	Afro-Asiatic	Hebrew	40 244	4862	4801	1263	158	158
Classical Chinese	lzh	Sino-Tibetan	Hanzi	346 778	43 067	43 323	68 991	8624	8624
Coptic	cop	Afro-Asiatic	Egyptian	57 493	7272	7558	1730	216	217
Gothic	got	Indo-European	Latin	44 044	5724	5568	4320	540	541
Medieval Icelandic	isl	Indo-European	Latin	473 478	59 002	58 242	21 820	2728	2728
Classical & Late Latin	lat	Indo-European	Latin	188 149	23 279	23 344	16 769	2096	2097
Medieval Latin	latm	Indo-European	Latin	599 255	75 079	74 351	30 176	3772	3773
Old Church Slavonic	chu	Indo-European	Cyrillic	159 368	19 779	19 696	18 102	2263	2263
Old East Slavic	orv	Indo-European	Cyrillic	250 833	31 078	32 318	24 788	3098	3099
Old French	fro	Indo-European	Latin	38 460	4764	4870	3113	389	390
Vedic Sanskrit	san	Indo-European	Latin (transcr.)	21 786	2729	2602	3197	400	400
Old Hungarian	ohu	Finno-Ugric	Latin	129 454	16 138	16 116	21 346	2668	2669

Table 9: Dataset statistics.