Allen Institute for AI @ SIGTYP 2024 Shared Task on Word Embedding Evaluation for Ancient and Historical Languages

Lester James V. Miranda
Allen Institute for Artificial Intelligence
ljm@allenai.org

Abstract

In this paper, we describe Allen AI's submission to the constrained track of the SIGTYP 2024 Shared Task. Using only the data provided by the organizers, we pretrained a transformer-based multilingual model, then finetuned it on the Universal Dependencies (UD) annotations of a given language for a downstream task. Our systems achieved decent performance on the test set, beating the baseline in most language-task pairs, yet struggles with subtoken tags in multiword expressions as seen in Coptic and Ancient Hebrew. On the validation set, we obtained ≥70% F1-score on most language-task pairs. In addition, we also explored the cross-lingual capability of our trained models. This paper highlights our pretraining and finetuning process, and our findings from our internal evaluations.

1 Introduction

This paper describes Allen AI's submission to the *constrained* track of the SIGTYP 2024 Shared Task on Word Embedding Evaluation for Ancient and Historical Languages. The constrained track requires participants to build a system for three linguistic tasks—parts-of-speech (POS) tagging, morphological annotation, and lemmatisation—using only the corpora provided by the organizers (Dereza et al., 2024).

The dataset contains Universal Dependencies v2.12 data (Zeman et al., 2023) in eleven languages with five Old Hungarian codices (HAS Research Institute for Linguistics, 2018). The texts were from before 1700 CE, containing four language families (Indo-European, Afro-Asiatic, Sino-Tibetan, and Finno-Ugric), and six scripts (Greek, Hebrew, Hanzi, Coptic, Latin, Cyrillic). Finally, the dataset has over 2.6M tokens for training, and around 330k tokens for validation and testing. Table 1 shows all languages for the subtask with their equivalent language code.

Language	Code
Old Church Slavonic	CHU
Coptic	COP
Old French	FRO
Gothic	GOT
Ancient Greek	GRC
Ancient Hebrew	HBO
Medieval Icelandic	ISL
Classical and Late Latin	LAT
Medieval Latin	LATM
Classical Chinese	LZH
Old Hungarian	OHU
Old East Slavic	ORV
Vedic Sanskrit	SAN

Table 1: Language codes for all thirteen languages in the constrained track of the shared task. We will refer to the language code in the succeeding tables and figures.

Our general approach involves pretraining a transformer-based multilingual language model (LM) on the shared task dataset (Dereza et al., 2024), and then finetuning the pretrained model using the Universal Dependencies (UD) annotations of each language. Throughout this paper, we will refer to the pretrained model as LIBERTUS. We also explored data sampling and augmentation techniques during the pretraining step to ensure better generalization performance.

On the validation set, we obtained ≥70% F1-score for the majority of language-task pairs. Moreover, our systems achieved decent performance on the test set, yet struggled with subtoken tags in multiword expressions as seen in Coptic (COP) and Ancient Hebrew (HBO). Table 2 shows our performance on the shared task test set. Our system achieved better than baseline performance in most language-task pairs, especially in morphological annotation, but underperformed in lemmatisation. Finally, our system ranked 2nd in the leaderboard

Lang.	POS tag.	Morph. annot.	Lemma.
CHU	0.946	0.941	0.796
COP	0.426	0.805	0.463
FRO	0.851	0.941	0.833
GOT	0.934	0.940	0.908
GRC	0.935	0.965	0.883
HBO	0.273	0.712	0.618
ISL	0.939	0.948	0.946
LAT	0.924	0.933	0.923
LATM	0.944	0.980	0.972
LZH	0.818	0.860	1.000
OHU	0.944	0.946	0.700
ORV	0.912	0.922	0.784
SAN	0.873	0.900	0.832

Table 2: SIGTYP 2024 Shared Task final leaderboard results as evaluated on the test set. Cells in green indicate scores that are greater than the baseline. The mapping of language codes to languages can be found in Table 1.

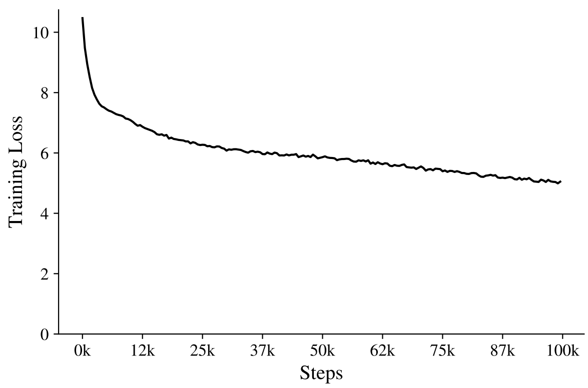

Figure 1: Training loss curve for the 126M-parameter model after 100k steps.

against four other submissions.

We detail our resource creation, model pretraining, and finetuning methodologies in this paper. The source code for all experiments can be found on GitHub: https://github.com/ljvmiranda921/LiBERTus.

2 Methodology

2.1 Model Pretraining

The main purpose of pretraining is to obtain context-sensitive word embeddings that we will finetune further for each downstream task. We approach this by training a multilingual language model akin to the XLM-RoBERTa (Conneau et al., 2020) and multilingual BERT (Devlin et al., 2019) architectures.

Hyperparameters	Value
Hidden size	768
Intermediate size	3072
Max position embed.	512
Num. attention heads	12
Hidden layers	12
Dropout	0.1

Table 3: Hyperparameter configuration for the LiBERTus pretrained model.

Preparing the pretraining corpora. We constructed the pretraining corpora using the annotated tokens of the shared task dataset. Initially, we explored several data augmentation techniques to ensure that each language is properly represented based on the number of unique tokens. However, we found pretraining to be unstable when we upsampled tokens to achieve the same count as Medieval Latin (LATM), the most overrepresented language. In the end, we found that leaving the token distribution as-is leads to more stable pretraining and lower validation scores. More information about our sampling experiments can be found in Section A.1 of the appendix.

Pretraining the base model. Using the pretraining corpora, we trained a model with 126M parameters that will serve as a base for finetuning downstream tasks. LiBERTus follows RoBERTa's pretraining architecture (Liu et al., 2019) and takes inspiration from Conneau et al. (2020)'s work on scaling BERT models to multiple languages.

Our hyperparameter choices closely resemble that of the original RoBERTa implementation as seen in Table 3. We also trained the same BPE tokenizer (Sennrich et al., 2016) using the constructed corpora. During model pretraining, we used the AdamW optimizer with β_2=0.98 and a weight decay of 0.01. The base model underwent training for 100k steps with a learning rate of 2e-4. We used a learning rate scheduler that linearly warms up during the first 12k steps of the training process, then linearly decays for the rest. Figure 1 shows the training curve.

2.2 Model Finetuning

For each language, we finetuned a multitask model using spaCy (Honnibal et al., 2020). We used spaCy's tokenization rules for the majority of languages except for Classical Chinese (LZH), where we segmented on characters. The final system consists of a parts-of-speech (POS) tagger, morphological analyzer, and lemmatizer.

Parts-of-speech (POS) tagger. We employed a standard classifier that predicts a vector of tag probabilities for each token. Each POS tag is a unique class that we assign exclusively to a token. We trained a model by taking the context-sensitive vectors from our pretrained embeddings, and passing it to a linear layer with a softmax activation. The network is then optimized using a categorical cross-entropy loss. For languages with subtokens such as Coptic (COP) and Ancient Hebrew (HBO), we merged each subtoken and used the full multi-word expression (MWE) during training.

Morphological analyzer. Similar to the POS tagger, we treat morphological annotation as a token classification task. Instead of directly modeling each feature, we made every unique combination of morphological features as a class. The limitation of this approach is that it can only predict combinations that were present in the training corpora. Similar to the POS tagger, we merged each subtoken for every multi-word expression (MWE) during training.

Lemmatizer. We trained a neural-based edit tree lemmatizer (Müller et al., 2015) by first extracting an edit tree for each token-lemma pair. Because this process can result to hundreds of edit trees, we treat the problem of picking the correct tree as a classification task. Here, each unique tree serves as a class and we compute a probability distribution over all trees for a given token. To obtain the most probable tree, we passed the context-sensitive embeddings from our pretrained model to a softmax layer and trained the network with a cross-entropy loss objective. We set the minimum frequency of an edit tree to 3, and used the surface form of the token as a backoff when no applicable edit tree is found. Finally, we ensured that the lemmatizer checks at least a single tree before resorting to backoff.

Finetuning the pipelines. We trained each component of the system in parallel, although the final "pipeline" assembles them together using the spaCy framework. For all components, the pretrained embeddings are passed on to linear layer with softmax activation. Sometimes, the tokenization from the multilingual model does not align one-to-one with spaCy's tokenization. In such case, we use a pooling layer that computes the average of each feature to obtain a single vector per spaCy token.

Lang.	POS tag.	Morph. annot.	Lemma.
CHU	0.947	0.876	0.803
COP	0.924	0.846	0.776
FRO	0.890	0.912	0.844
GOT	0.951	0.886	0.914
GRC	0.956	0.915	0.873
HBO	0.624	0.561	0.219
ISL	0.963	0.901	0.949
LAT	0.949	0.882	0.922
LATM	0.984	0.951	0.968
LZH	0.795	0.824	0.942
OHU	0.953	0.919	0.697
ORV	0.933	0.859	0.787
SAN	0.888	0.811	0.817

Table 4: F1-score results on the validation set. The mapping of language codes to languages can be found in Table 1.

During finetuning, we used the Adam optimizer with β_1=0.9, β_2=0.999 and a learning rate of 0.001. The learning rate warms up linearly for the first 250 steps, and then decays afterwards.

3 Results

Table 2 shows the test scores for the shared task using the official shared task metrics:

- **POS-tagging:** Accuracy@1, F1-score
- **Detailed morphological annotation:** Macro-average of Accuracy@1 per tag
- **Lemmatisation:** Accuracy@1, Accuracy@3

Our systems obtained decent performance and beat the baseline for the majority of language-task pairs. In addition, our submission obtained 2nd place against three other submissions.

In the following sections, we will outline our internal evaluations and benchmarking experiments.

3.1 Performance on the validation set

Table 4 shows the validation scores of our finetuned models. We achieved $\geq$70% performance in most language-task pairs. The top performers, calculated by taking the average across all tasks, are Medieval Latin (0.968), Medieval Icelandic (0.938), and Classical and Late Latin (0.918), whereas the

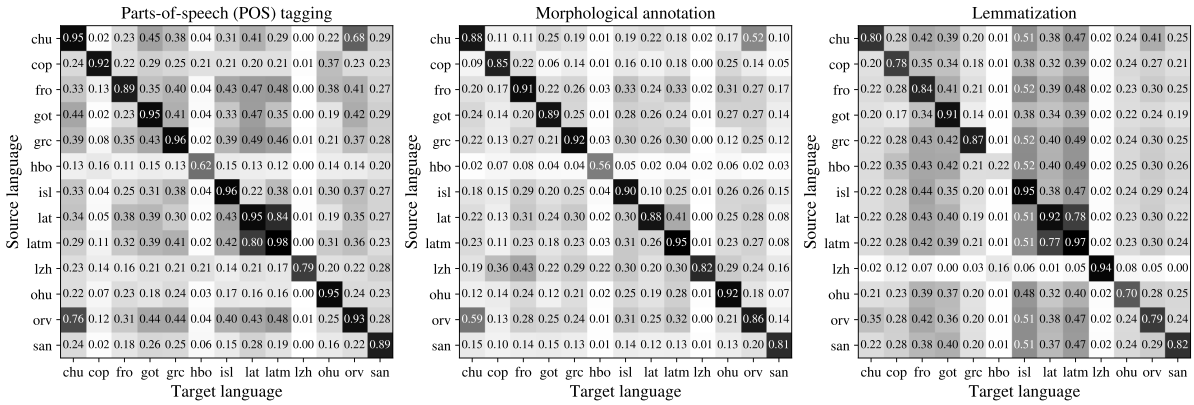

Figure 2: Cross-lingual evaluation, as measured by the F1-score, given a monolingual model from one language and a validation set in another. The mapping of language codes to languages can be found in Table 1.

bottom performers are Coptic (0.849), Vedic Sanskrit (0.839), and Ancient Hebrew (0.468).

Compared to our validation scores, our leaderboard scores on Coptic (COP) and Ancient Hebrew (HBO) are poor. This performance is due to our models being unable to accurately predict subtoken information as it has only seen the full MWE during training. In order to align our tokenization with the shared task's validation script in Codalab (Pavao et al., 2023), we substituted each MWE with its subtokens resulting to potentially incomprehensible text. Finally, for empty tokens such as previously found in Old East Slavic (ORV), we added a rule in our system to produce empty predictions.[1]

3.2 Evaluating cross-lingual capabilities

To test the cross-lingual capability of a language, we evaluated its finetuned model to the validation set of another. Figure 2 shows the results.

We found that it is possible to adapt a language onto another for morphological annotation and lemmatization. However, this does not extend to its morphology, as the validation set performs best only in the language it was trained on.

Some target languages tend to be cross-lingually receptive on lemmatization, i.e., many source languages can perform decently when applied to them. This pattern is apparent especially between Classical and Late Latin (LAT) and Medieval Latin (LATM) due to the latter being a direct continuation of the former. In addition, we also observed that Old Church Slavonic (CHU) and Old East Slavic (ORV) are also cross-lingually compatible as they came from the same Indo-European root. Finally, we noticed that languages coming from the same family and script such as Old French (FRO), Gothic (GOT), Medieval Icelandic (ISL), Classical and Late Latin (LAT), and Medieval Latin (LATM) tend to have good cross-lingual transfer capabilities.

[1]This problem was initially caused by missing brackets in the reference annotations. We used the correct tokens for ORV in the final submission.

4 Related Work

Multilingual language modeling. Several efforts in the field of NLP involve pretraining a transformer network (Vaswani et al., 2017) using a corpus of multiple languages such as XLM-RoBERTa (Liu et al., 2019) and multilingual BERT (Conneau et al., 2020). The main advantage of multilingual language modeling is that they can harness the cross-lingual representations of the languages in its corpora to solve several downstream tasks (Chang et al., 2022). However, these models harness modern data sources, not ancient and historical text. The only exemplar that we found for a historical and multilingual LM is hmBERT (Schweter et al., 2022), where they pretrained on English, German, French, Finnish, Swedish, and Dutch languages from 1800 onwards. LIBERTUS aims to extend this literature by providing another multilingual language model focusing on ancient and historical texts from diverse scripts and language families before 1700 CE.

5 Conclusion

This paper describes Team Allen AI's system: a pretrained multilingual model (LIBERTUS) finetuned on different languages for each downstream task. Our system obtained decent performance for the majority of language-task pairs. However, due to our training paradigm, it struggles annotating subtokens of multiword expressions. Nevertheless, our validation scores are high (≥70% F1-score) for the majority of language-task pairs.

We also evaluated each language's cross-lingual capability and showed that transfer learning is possible especially on lemmatization. This approach can be a viable alternative on limited corpora.

Our training and benchmarking source code is on GitHub: `https://github.com/ljvmiranda921/LiBERTus`. The pretrained multilingual model and finetuned pipelines are also available on HuggingFace.[2]

Limitations

Pretrained LM size. Due to compute constraints, we were only able to pretrain a model akin to the size of $\text{RoBERTa}_{\text{base}}$. We highly recommend pretraining a large LIBERTUS model to obtain performance gains if the resource allows.

Pretraining data mix. In the end, we didn't employ any sampling strategy to balance the token distribution of different languages during pretraining. We only tested simple up-/downsampling strategies and our experiments are limited to repeating available data.

Label combination as individual classes. When training the morphologizer and POS tagger, we treated each feature and parts-of-speech combination as its own class instead of modeling them individually. This limits our text classifier to only predicting combinations it has seen during the training process.

Subtoken performance for multiword expressions. Our systems performed poorly on COP and HBO in the leaderboard due to how we trained our model. Instead of showing subtokens, we used the full multi-word expression during training.

[2] `https://huggingface.co/collections/ljvmiranda921/sigtyp2024-shared-task-models-65629ea0462e5ebcbf1a2133`

References

Burton H. Bloom. 1970. Space/time trade-offs in hash coding with allowable errors. *Commun. ACM*, 13(7):422–426.

Tyler Chang, Zhuowen Tu, and Benjamin Bergen. 2022. The geometry of multilingual language model representations. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 119–136, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised Cross-lingual Representation Learning at Scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Oksana Dereza, Adrian Doyle, Priya Rani, Atul Kr. Ojha, Pádraic Moran, and John P. McCrae. 2024. Findings of the SIGTYP 2024 shared task on word embedding evaluation for ancient and historical languages. In *Proceedings of the 6th Workshop on Research in Computational Linguistic Typology and Multilingual NLP (SIGTYP 2024)*, St. Julians, Malta. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

HAS Research Institute for Linguistics. 2018. Old Hungarian Codices. *Hungarian Generative Diachronic Syntax*. Accessed: October 22, 2023.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength Natural Language Processing in Python.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv:1907.11692*.

Lester James Miranda, Ákos Kádár, Adriane Boyd, Sofie Van Landeghem, Anders Søgaard, and Matthew Honnibal. 2022. Multi hash embeddings in spaCy.

Thomas Müller, Ryan Cotterell, Alexander Fraser, and Hinrich Schütze. 2015. Joint lemmatization and morphological tagging with lemming. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2268–2274, Lisbon, Portugal. Association for Computational Linguistics.

Adrien Pavao, Isabelle Guyon, Anne-Catherine Letournel, Dinh-Tuan Tran, Xavier Baro, Hugo Jair Escalante, Sergio Escalera, Tyler Thomas, and Zhen Xu. 2023. Codalab competitions: An open source platform to organize scientific challenges. *Journal of Machine Learning Research*, 24(198):1–6.

Stefan Schweter, Luisa März, Katharina Schmid, and Erion cCano. 2022. hmbert: Historical multilingual language models for named entity recognition. In *Conference and Labs of the Evaluation Forum*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

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, Phuong 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, Luong 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, Steinhór 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 Hórðarson, Vilhjálmur Horsteinsson, 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.

A Appendix

A.1 Different sampling strategies on pretraining validation performance

We explored different sampling strategies and their effect on the pretraining validation loss curve as shown in Figure 3. We ran the pretraining pipeline for 20k steps (one-fifths of the final hyperparameter value) and measured the validation loss. The evaluation corpus was built from the validation set of the shared task, and we kept it the same throughout the experiment. We tested the following sampling strategies:

- **None:** we used the original dataset without any data sampling or augmentation.
- **Upsampling:** we upsampled each language to ensure that the number of their unique tokens is greater than or equal to the most dominant language.

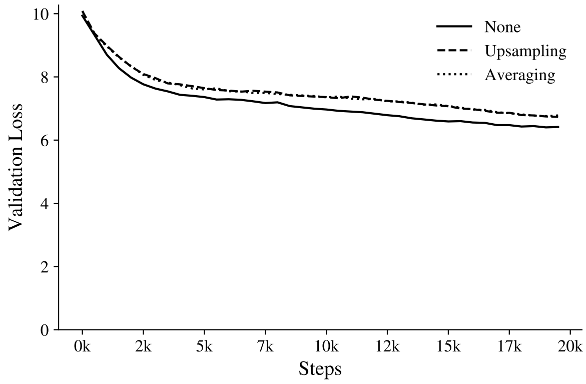

Figure 3: Validation loss curve for different sampling strategies in 20k steps.

- **Averaging:** we took the average number of unique tokens in the whole set and up-/downsampled each language based on this value.

Because any form of sampling resulted to unstable pretraining and higher validation loss, we decided to stick with the dataset's original data distribution. We highly recommend exploring alternative data mixes to ensure that all languages will be represented while keeping the training process stable.

A.2 Finetuning a model per language vs. monolithic system

We investigated if finetuning a model per language is more effective against a monolithic system, i.e., training on the full multilingual annotated corpora. Here, we combined the training corpora for all languages, then shuffled them before batching. The merged dataset has 194,281 documents for training and 26,954 documents for validation. This means that the downstream model sees a language mix per training epoch.

As shown in Figure 4, finetuning a model per language still yields the best results. One advantage of language-specific models is that we were able to set a different tokenizer per language—enabling us to get decent scores on Classical Chinese (LZH). Training the monolithic model is also sensitive to the training data distribution, as shown by the disparity in performance between majority languages (Classical and Late Latin, Medieval Latin) and minority ones (Old Hungarian, Old East Slavic, Vedic Sanskrit). Due to these findings, we decided to train multiple models for our final system.

A.3 Alternative approach—multi-hash embeddings

We considered using multi-hash embeddings (Miranda et al., 2022) as an alternative approach. Instead of pretraining, these embeddings use orthographic features (e.g., prefix, suffix, norm, shape) to create a word vector table. This approach also applies the hashing trick, inspired by Bloom filters (Bloom, 1970), to decrease the vector table's memory footprint.

Figure 5 shows the results in comparison to our final system. It is notable that simple orthographic features are competitive with our transformer-based model. However, we chose to submit the transformer-based pipeline as our final system because it still outperforms the multi-hash embed method in the majority of our language-task pairs. We still recommend investigating this approach further because the hash-embed method has noticeable efficiency gains in terms of model size.

A.4 Open-access models

Finetuned spaCy models. The finetuned models follow spaCy's naming convention and are available on HuggingFace (`https://huggingface.co/ljvmiranda921/{model_name}`, where `model_name` is found on Table 5). Table 5 also lists the size (in MB) for each language. Each model is multi-task in nature and can perform all downstream tasks (i.e., POS tagging, morphological annotation, lemmatisation) in a single call using the spaCy framework.[3]

Pretrained multilingual LM. In addition, the pretrained multilingual language model is also on HuggingFace (`https://huggingface.co/ljvmiranda921/LiBERTus-base`) and can be accessed using the `transformers` library (Wolf et al., 2020).

[3] `https://spacy.io`

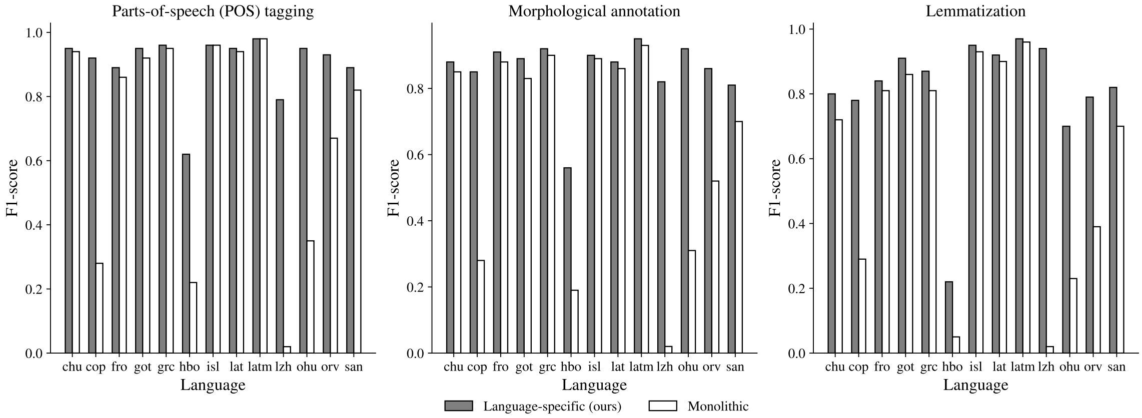

Figure 4: Comparison between training language-specific models versus a single monolithic model as evaluated on the validation set.

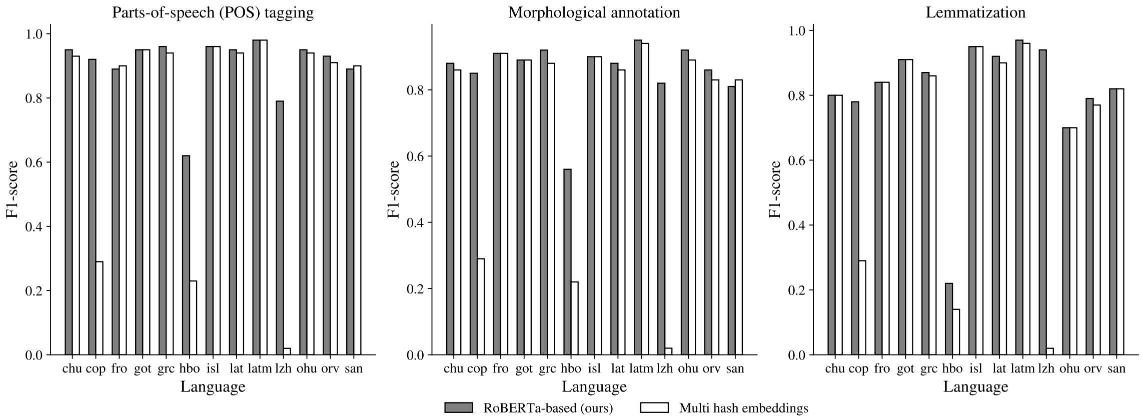

Figure 5: Comparison between a RoBERTa-based pretrained model and multi-hash embeddings (Miranda et al., 2022) as evaluated on the validation set.

Language	spaCy Model Name	Model Size
Old Church Slavonic	`xx_chu_sigtyp_trf`	491 MB
Coptic	`el_cop_sigtyp_trf`	477 MB
Old French	`xx_fro_sigtyp_trf`	471 MB
Gothic	`xx_got_sigtyp_trf`	474 MB
Ancient Greek	`xx_grc_sigtyp_trf`	501 MB
Ancient Hebrew	`he_hbo_sigtyp_trf`	475 MB
Medieval Icelandic	`xx_isl_sigtyp_trf`	496 MB
Classical and Late Latin	`xx_lat_sigtyp_trf`	486 MB
Medieval Latin	`xx_latm_sigtyp_trf`	510 MB
Classical Chinese	`zh_lzh_sigtyp_trf`	469 MB
Old Hungarian	`xx_ohu_sigtyp_trf`	488 MB
Old East Slavic	`xx_orv_sigtyp_trf`	503 MB
Vedic Sanskrit	`xx_san_sigtyp_trf`	473 MB

Table 5: The finetuned model for each language is available on HuggingFace.