BelaMorphLM: A Controlled Study of Morphology-Aware Tokenization for Belarusian Encoder Language Models

Anonymous ACL submission

Abstract

Tokenization is a central component of pretrained language models, especially for morphologically rich low-resource languages. We present a controlled study of morphology-aware tokenization for Belarusian masked language models. To isolate the effect of tokenization, we pretrain three RoBERTa-family models on the same Belarusian HPLT corpus, varying only the tokenizer: standard WordPiece, WordPiece trained on morpheme-segmented text, and a dictionary-based hybrid morpheme tokenizer with WordPiece fallback. We evaluate the models on masked language modeling, Universal Dependencies morphology-sensitive tasks, and BelarusianGLUE. Morphology-aware tokenization substantially improves morpheme-boundary preservation and yields clear gains on explicitly morphological tasks, especially fine-grained feature prediction and lemmatization, where models must recover information encoded in inflectional structure. However, these gains do not uniformly transfer to broad sentence-level NLU: on BelarusianGLUE, morphology-aware tokenizers improve some tasks but remain comparable to or weaker than standard WordPiece on others. These findings suggest that morpheme-informed tokenization is a targeted inductive bias rather than a universal replacement for statistical subword segmentation.

1 Introduction

While the quality of pretrained language models is largely shaped by model architecture, training data, and optimization, tokenization remains an important part of the modeling pipeline. It determines how text is decomposed into input units and, in masked language modeling, which units the model is trained to predict. Standard subword tokenizers such as BPE (Sennrich et al., 2016) and WordPiece (Schuster and Nakajima, 2012) are effective and language-agnostic, but their segments are optimized from corpus statistics and need not correspond to linguistically meaningful morphemes. This mismatch may be especially important for morphologically rich languages, where grammatical and lexical information is often encoded in affixes and endings.

Recent work has therefore revisited morphology-aware tokenization as a way to make subword units more linguistically meaningful (Park et al., 2020; Jabbar, 2024; Goot et al., 2025; Kildeberg et al., 2025; García et al., 2025; Držík and Kapusta, 2026). However, existing evidence remains incomplete: better morpheme-boundary preservation does not necessarily imply lower language-modeling loss or uniformly better downstream performance. In particular, it remains unclear whether morphology-aware tokenization improves broad NLU tasks, explicit morphological prediction, or only intrinsic tokenization metrics.

We study this question for Belarusian, a low-resource East Slavic language with rich inflectional and derivational morphology. Belarusian is a useful test case because the recent availability of large-scale Belarusian pretraining data in HPLT v2 (Burchell et al., 2025), a morphemic resource and high-quality segmentation approach for Belarusian (Morozov et al., 2025), and BelarusianGLUE (Aparovich et al., 2025) makes a controlled study possible.

To isolate the effect of tokenization, we conduct a controlled pretraining experiment: three RoBERTa-family masked language models are trained on the same Belarusian HPLT corpus (Burchell et al., 2025) and differ only in their tokenizer. The tokenizers represent three paradigms: standard WordPiece, WordPiece trained on morpheme-segmented text, and a dictionary-based hybrid morpheme tokenizer with WordPiece fallback. We then trace the effect of tokenization from morpheme-boundary preservation to masked language modeling, morphological

feature prediction, and general NLU tasks. Our results show a selective effect: morphology-aware tokenization improves explicitly morphological tasks, especially feature prediction and lemmatization, but does not uniformly improve broad BelarusianGLUE tasks.

2 Related Work

Subword tokenization is the standard open-vocabulary solution in neural NLP. BPE, WordPiece, SentencePiece, and Unigram tokenization made it possible to model rare and unseen words with fixed-size vocabularies and became default components of pretrained Transformers (Schuster and Nakajima, 2012; Kudo and Richardson, 2018; Kudo, 2018). However, these tokenizers optimize corpus statistics rather than linguistic structure, and their segments often do not correspond to morpheme boundaries. This may be problematic for morphologically rich languages, where grammatical and lexical information is regularly encoded inside word forms.

Earlier work on subword and character-based representations showed that such models can capture some morphological regularities, but also that they do not fully replace access to explicit morphological analyses (Vania and Lopez, 2017). More recent work directly comparing tokenization methods for masked language model pretraining has shown, for English and Japanese, that BPE is less aligned with morphological reference segmentations than Unigram LM tokenization, and that the latter can match or outperform BPE in downstream evaluation (Bostrom and Durrett, 2020). Cross-linguistic evidence from LSTM language modeling further suggests that morphology interacts with modeling difficulty: under BPE segmentation, several measures of morphological complexity correlate with higher surprisal, whereas linguistically motivated segmentation methods such as Morfessor and finite-state analyzers can reduce this effect (Park et al., 2021).

These observations have motivated a growing line of work on morphology-aware tokenization. For Korean, Park et al. (2020) show that applying morphological segmentation before BPE gives the best results on Korean–English machine translation and several NLU tasks, although standard BPE remains strongest for machine reading comprehension. MorphPiece combines a morpheme lookup table with BPE fallback and reports comparable or superior performance of a GPT-style model over GPT-2 on language modeling, GLUE-style zero-shot evaluation, and embedding benchmarks (Jabbar, 2024). Recent work on Danish similarly shows that morphology-aware tokenizers substantially improve morpheme-boundary preservation and can improve downstream performance of generative transformer models (Goot et al., 2025; Kildeberg et al., 2025). For Slovak, Držík and Kapusta (2026) report modest but consistent gains from a morphology-aware BPE variant over standard BPE in RoBERTa-style MLM pretraining. Their downstream evaluation includes NER, UPOS tagging, sentiment and topic classification, extractive QA, and STS, with the strongest improvements in semantic similarity and question answering. However, the morphology-oriented evaluation is limited to UPOS tagging and does not cover lemmatization or fine-grained morphological feature prediction. For Spanish, García et al. (2025) report that a BPE tokenizer trained on morphologically segmented text outperforms standard BPE in masked language modeling and in a limited downstream evaluation covering natural language inference, paraphrase identification, and semantic textual similarity. At the same time, the authors note that they do not perform a detailed evaluation on morphologically demanding tasks such as lemmatization or POS tagging. Overall, the existing evidence points to potential benefits of morphology-aware tokenization, but these benefits are uneven across tasks and sometimes absent. At the same time, the most intuitive potential advantage — improvements on morphologically demanding tasks such as lemmatization or morphological tagging — remains substantially underexplored.

3 Morpheme Segmentation

Morpheme segmentation provides the linguistic signal used to construct morphology-aware tokenizers. Since morphemic dictionaries rarely cover all inflected and derived forms found in large corpora, dictionary lookup must be complemented by automatic segmentation. For Slavic languages, both convolutional networks and BERT-like models have been shown to achieve strong segmentation quality (Morozov et al., 2025).

3.1 Slounik-WF

The starting point for our segmentation pipeline is the *Slounik* dataset[1] (Morozov et al., 2025), a Belarusian morpheme segmentation resource derived from the *School Morpheme Dictionary of the Belarusian Language* (Mormysh et al., 2005). The resource follows a surface segmentation scheme: morphemes are represented as substrings of the original word form and are not normalized to underlying or citation forms. Each segment is assigned a morpheme type, such as PREF, ROOT, SUFF, END, or LINK. This representation is well suited for morphology-aware tokenization, because the tokenizer must operate over surface strings that can be concatenated back into the original corpus word.

However, the original resource is not sufficient for large-scale tokenizer construction. Like many morphemic dictionaries, it mainly covers dictionary forms and selected entries, whereas pretraining corpora contain a much larger variety of inflected and derived word forms. We therefore use an expanded word-form-level version of the resource, *Slounik-WF*. Slounik-WF extends Slounik from lemma-level segmentations to 332k word-form segmentations using generated inflectional paradigms, heuristic segmentation transfer, and manual validation. The expansion relies on the Grammatical Database of the Belarusian Language[2] and generates inflected forms for nouns, adjectives, and verbs, while preserving adverbs as uninflected forms. The expansion procedure details are presented in Appendix F.

3.2 Evaluation of Segmentation Models

Before using automatic segmentation for tokenizer construction, we evaluated several segmentation models on *Slounik-WF*. This step is important because the downstream comparison of tokenizers depends on the quality of the morpheme signal: if segmentation errors are frequent, the effect of morphology-aware tokenization cannot be interpreted reliably.

We compare three approaches that have shown strong results in surface morpheme segmentation: convolutional neural networks, LSTM networks, and pretrained encoder models. The CNN baseline follows a character-level ResNet architecture, while the LSTM baseline uses a character-level BiLSTM (Olbrich and Žabokrtský, 2026). Both models treat segmentation as character-level sequence labeling, where each character receives a label encoding its position inside a morpheme and the corresponding morpheme type. For pretrained encoders, we fine-tune BERT-like models in the same sequence-labeling setting (Morozov et al., 2025). The input sequence consists of the word split into characters followed by the complete word; labels assigned to complete-word tokens are ignored in the loss and in metric computation. The training and fine-tuning hyperparameters are reported in Appendix G.

To assess the segmentation quality, we used 5-fold cross-validation. To evaluate generalization, we use three splitting scenarios. In the *split-by-form* setting, word forms are randomly assigned to folds. In the *split-by-lemma* setting, all forms of the same lemma are assigned to the same fold, testing whether the model can segment unseen paradigms. In the *split-by-root* setting, words are grouped by root, and roots do not overlap between training and test folds; this is the most difficult scenario and tests generalization to unseen lexical morphemes. Performance is reported in Table 1 using average morpheme-boundary F1 and word-level accuracy.

split-by-...	**form**		**lemma**		**root**	
	F1	WA	F1	WA	F1	WA
CNN	**99.9**	98.8	98.4	90.3	94.3	68.7
LSTM	**99.9**	98.9	98.4	89.8	94.3	67.6
belRoberta	99.8	98.4	97.9	87.7	94.6	70.2
HPLT be	99.4	95.3	98.1	87.9	95.9	76.3
ruBert	**99.9**	**99.2**	98.4	90.1	94.7	70.8
ruRoberta	**99.9**	98.8	98.5	90.5	95.3	73.2
SlavicBERT	99.7	97.4	98.0	87.8	94.8	70.5
mBERT	**99.9**	**99.2**	98.5	90.9	94.9	71.7
distilBERT	**99.9**	99.1	98.4	90.2	94.8	70.9
XLMR-large	**99.9**	99.1	**98.8**	**92.5**	**96.2**	**77.7**

Table 1: Performance comparison of different segmentation models. Bold indicates the best result. F1 stands for F1-score for morpheme boundaries, WA for word-level accuracy; belRoberta = KoichiYasuoka/roberta-small-belarusian, HPLT be = HPLT/hplt_bert_base_2_0_bel-Cyrl, ruBert = ai-forever/ruBert-base, ruRoberta = ai-forever/ruRoberta-large, slavicBERT = DeepPavlov/bert-base-bg-cs-pl-ru-cased, mBERT = google-bert/bert-base-multilingual-cased, distilBERT = distilbert/distilbert-base-multilingual-cased, XLMR-large = FacebookAI/xlm-roberta-large (see Appendix B).

The results show that the random split is almost saturated: several models reach near-perfect boundary F1 and word-level accuracy above 99%. The more informative settings are the lemma- and

[1] https://huggingface.co/datasets/ruscorpora/morphodict-bel

[2] https://github.com/Belarus/GrammarDB

root-based splits. In these scenarios, fine-tuning pretrained encoders appears to be more effective than the CNN and LSTM baselines. The best overall model is XLM-RoBERTa-large, which reaches approximately 92.5% word-level accuracy in the lemma-based split and 77.7% in the root-based split.

4 Model Pretraining

To isolate the effect of tokenization, we pretrain three RoBERTa-family masked language models under matched conditions. The models share the same architecture, vocabulary size, pretraining corpus, and training budget, and differ only in the tokenizer used to convert text into model inputs. All models are trained with the masked language modeling (MLM) objective, following the RoBERTa-style setup (Liu et al., 2019).

4.1 Tokenizers

We compare three tokenization strategies with the same vocabulary size of 50,000 and the same special tokens. T1 is a standard WordPiece tokenizer trained on the raw Belarusian HPLT corpus. T2 is a WordPiece tokenizer trained on a morpheme-segmented version of the same corpus: the morphemes are written as whitespace-separated units before tokenizer training, encouraging WordPiece merges to respect morpheme boundaries. T3 is a dictionary-based hybrid tokenizer: words covered by the morphemic dictionary are segmented into morphemes, while uncovered words are encoded with a fallback WordPiece tokenizer.

The three tokenizers differ substantially in their relation to morpheme boundaries. The standard WordPiece tokenizer produces the shortest token sequences, with approximately 1.34 tokens per word on the validation and test splits. In contrast, T2 and T3 produce approximately 2.77–2.78 tokens per word. This higher token fertility is expected: the morpheme-aware tokenizers represent many words as sequences of roots, suffixes, endings, and other morphemes. However, the increase in sequence length is accompanied by much stronger alignment with linguistic segmentation. In our intrinsic tokenizer analysis, T2 and T3 achieve morpheme-boundary F1 above 99%, while the standard WordPiece baseline remains close to 10%.

For T3, we additionally measured the unknown-token rate after applying the tokenizer to all corpus splits. Initial versions of the tokenizer produced a high number of unknown tokens due to multi-character punctuation sequences and rare non-word symbols. After extending the required symbol inventory and improving non-word token handling, the final T3 tokenizer achieved a low UNK rate of 0.059% on the training split, 0.062% on validation, and 0.062% on test.

4.2 Pretraining Corpus

We use the Belarusian subset of HPLT v2.0[3] as the pretraining corpus. HPLT v2.0 is a large multilingual collection of monolingual and parallel corpora; its monolingual component contains approximately 8 trillion tokens across 193 languages (Burchell et al., 2025).

The cleaned Belarusian file contains 42.95 million lines and approximately 1.16 billion whitespace-delimited tokens. Because HPLT is web-derived and contains many duplicated fragments, we deduplicate it at the line level. After filtering and splitting, the corpus contains 21.05M training lines, 214.5k validation lines, and 213.8k test lines. The same split is used for all three tokenizers.

To extend the coverage of our morpheme-annotated lexicon, we applied the XLM-RoBERTa-large segmentation model described in the Section 3 to out-of-dictionary word forms from our pretraining corpus. We first extracted all word forms occurring in the training split that were not covered by Slounik-WF and satisfied three filtering criteria: frequency of at least three occurrences, length between 3 and 50 characters, and exclusive use of Cyrillic characters. In total, this filtering selected 1,469,980 word forms, covering approximately three quarters of all word tokens in the training data by frequency. We segmented these word forms with the trained model and merged the predicted segmentations with the original analyses from Slounik-WF, obtaining an extended morpheme-segmented resource, *Slounik-WF-ext*, used in the subsequent tokenizer construction pipeline. The Slounik-WF-ext dictionary covers approximately 79% of word tokens in all three splits: 79.01% in training, 78.87% in validation, and 78.92% in test. This stable coverage across splits suggests that the validation and test data are comparable to the training data with respect to dictionary availability.

For T1, the raw split files are tokenized dir-

[3] https://hplt-project.org/datasets/v2.0

ectly. For T2, we apply WordPiece tokenizer on the morpheme-segmented version of the corpus splitted using Slounik-WF-ext. For T3, we apply the hybrid tokenizer directly to the raw text, using dictionary segmentation for covered word forms and WordPiece fallback otherwise. In all cases, the resulting datasets store token IDs, attention masks, and word-to-token mappings required for whole-word masking.

4.3 Pretraining Setup

All three models use the same RoBERTa-family encoder architecture and are trained from scratch. We use a base-size configuration with 12 Transformer layers, hidden size 768, 12 attention heads, intermediate size 3072, and a vocabulary of 50,000 tokens (for details, see Appendix H). The maximum sequence length is set to 512. Special token IDs are kept identical across tokenizers: `[PAD]`, `[UNK]`, `[CLS]`, `[SEP]`, and `[MASK]`. The models differ only in the tokenizer and in the resulting tokenized training data. All pretraining runs are performed under the same computational setup. We train each model on a single multi-GPU node with 10 NVIDIA A100 GPUs using distributed data parallel training.

We train the models with the MLM objective. Each training example is a packed token sequence with dynamic masking applied during training. We use a masking probability of 15%. To make the masking procedure comparable across tokenizers, we apply whole-word masking: if a word is selected for masking, all tokens originating from the same whitespace-delimited word are masked.

5 Evaluation

We evaluate the models at three levels: intrinsic MLM performance, morphology-aware tasks on Universal Dependencies (UD) treebank (Nivre et al., 2020), and broad NLU on Belarusian-GLUE (Aparovich et al., 2025). The intrinsic evaluation checks whether each model learns its masked-token objective, while UD tests the central hypothesis that morpheme-aware tokenization helps when predictions depend on word-internal grammatical information. BelarusianGLUE then tests whether such effects transfer to general NLU.

5.1 Intrinsic Evaluation

We first evaluate the pretrained models on the MLM objective using the validation and test splits of the pretraining corpus. For each model, we apply the same whole-word masking procedure as during pretraining and report cross-entropy loss, top-1 and top-5 masked-token accuracy, and whole-word exact match. The latter metric counts a masked word as correct only if all of its supervised tokens are predicted correctly.

Tokenizer	Loss	Acc@1	Acc@5	WW exact
T1	3.3026	45.92	61.46	48.81
T2	3.5750	46.55	64.81	40.62
T3	3.6608	46.05	64.16	40.54

Table 2: Intrinsic MLM evaluation on the test split. Accuracy and whole-word exact match are reported in percent.

Because the three tokenizers define different prediction units, the MLM loss should not be interpreted as a direct measure of overall model quality across tokenizers. In particular, T1 predicts standard WordPiece units, whereas T2 and T3 often predict morpheme-level units. The intrinsic evaluation is therefore used primarily as a sanity check and as a way to characterize how the different tokenization schemes affect the MLM objective.

The results in Table 2 show that T1 obtains the lowest loss, which is expected because it predicts coarser, frequency-optimized WordPiece units. T2 and T3 obtain higher Acc@5 but lower whole-word exact match; the latter is stricter for morpheme-aware tokenizers because more tokens must be recovered for each masked word. We therefore treat MLM results as diagnostic rather than as a direct ranking.

5.2 Morphology-aware Evaluation

To test whether morphology-aware tokenization improves tasks that require access to word-internal grammatical information, we evaluate the models the UD Belarusian HSE treebank[4], based on news, fiction, and social media texts from the Belarusian–Russian parallel subcorpus of the Russian National Corpus (Savchuk et al., 2024). We use the official train/dev/test split.

We formulate two token classification tasks. In the first task, the model predicts the universal part-of-speech tag (UPOS) for each word. In the second task, the model predicts the full UD morphological feature bundle (FEATS) as a single flat label. Thus, a bundle such as `Case=Nom|Gender=Fem|Number=Sing` is treated as one class. Feature bundles not observed in the

[4] https://github.com/UniversalDependencies/UD_Belarusian-HSE

training split are mapped to a special __UNK__ label for the FEATS task.

In addition to UPOS and FEATS prediction, we evaluate the models on contextual lemmatization (LEMMA). We use rule classification: each label is a suffix-replacement rule derived from the longest common prefix between the surface form and lemma (see Table 3). Rules observed in the training split define the label inventory; gold rules in development or test that are not present in training are mapped to a special unknown rule. We report lemma accuracy and changed-form accuracy, the latter computed only for tokens whose surface form differs from the lemma.

For all tokenizers, labels are assigned only to the first token corresponding to a word; all remaining tokens of the same word are ignored in the loss. For T2 and T3, the same morpheme dictionary used in tokenizer construction is used to reproduce the corresponding word-to-token alignment during fine-tuning.

For each task, we fine-tune each pretrained model with a token classification head for 20 epochs. We use a learning rate of 5×10^{-5}, weight decay of 0.01, warmup ratio of 0.06, per-device training batch size of 16, per-device evaluation batch size of 32, maximum sequence length of 512, and bfloat16 precision. To account for fine-tuning variability, each experiment is repeated with 15 random seeds. We report mean and standard deviation over seeds.

Table 4 summarizes the morphology-aware UD evaluation. The effect of tokenization is clearly task-dependent. For coarse UPOS tagging, all three models perform similarly: accuracy is above 98%, and the support-filtered macro-F1 score, excluding the single `INTJ` token, differs by less than one point. This suggests that coarse part-of-speech categories are largely recoverable from lexical and contextual information and do not strongly benefit from explicit morpheme-aware segmentation. Additional details are provided in Appendix I.1.

The pattern changes for tasks that require more direct access to word-internal morphology. On FEATS, both morphology-aware tokenizers improve exact bundle accuracy by about 1.1 points over standard WordPiece and improve feature-value micro-F1 by about 0.5 points. The macro-F1 gain is smaller for T2 but larger for T3, indicating that the benefit is not restricted to frequent feature bundles, although rare bundles remain unstable. Lemmatization shows an even stronger effect: T2 and T3 improve lemma accuracy by approximately 1.7 and 1.6 points, respectively, and the gain rises to more than 3.4 points on changed forms. This suggests that the improvement is not due to copying tokens that are already in lemma form, but to better handling of non-trivial inflectional transformations. To further illustrate how explicit morpheme segmentation aids the model in recovering complex inflectional traits compared to purely statistical subword splits, we provide a qualitative error analysis and specific prediction examples in Appendix I.3.

To better understand which grammatical categories benefit from morphology-aware tokenization, we further decompose the predicted feature bundles into individual feature-value decisions. Table 5 aggregates individual features into four linguistically motivated groups. The clearest improvements are observed for core nominal and verbal morphology. In the nominal core group, which includes number, case, gender, animacy, and degree, T2 and T3 improve over T1 by approximately 0.5 F1 points. In the verbal core group, the gains are larger: T2 improves over T1 by 0.8 points, and T3 by 0.9 points. The effect is weaker for pronominal and quantificational features, where all three models are very close. For the lexical residual group, which contains low-support or more lexicalized features, morphology-aware tokenization does not provide a consistent advantage. This suggests that the benefits of morpheme-aware tokenization are selective rather than universal: they are most visible in core inflectional morphology.

The full per-feature results are reported in Appendix I.2. The largest gains are observed for `Gender` (+1.8 for T2 and +1.7 for T3 over T1), `Voice` (+1.1/+1.1), `Aspect` (+0.8/+1.1), `Tense` (+0.9/+0.9), and `Person` (+0.6/+0.7). These categories are closely tied to Belarusian inflectional morphology. By contrast, T1 remains slightly better for `Case` and for several low-support or lexicalized features, including `Abbr`, `Foreign`, `Variant`, and `InflClass`.

Overall, morpheme-aware tokenization consistently helps on tasks where the prediction depends on morphological structure: full morphological feature prediction and lemmatization. The two morphology-aware strategies behave similarly, with T2 slightly ahead on lemmatization and T3 slightly ahead on macro-F1 for FEATS.

Rule	Supp.	Examples
COPY	146k	Карэя→Карэя (PROPN, *'Korea'*); інвеставаць→інвеставаць (VERB, *'to invest'*)
del 1	21k	візіту→візіт (NOUN, *'visit'*); сакавіка→сакавік (NOUN, *'March'*)
del 1, + -а	9k	эканоміку→эканоміка (NOUN, *'economy'*); міністэрстве→міністэрства (NOUN, *'ministry'*)
del 2	6k	інвестыцыйнымі→інвестыцыйны (ADJ, *'investment'*); ходзе→ход (NOUN, *'course'*)
del 2, + -ць	5k	падпісала→падпісаць (VERB, *'to sign'*); растлумачылі→растлумачыць (VERB, *'to explain'*)
del 2, + -ы	5k	юрыдычная→юрыдычны (ADJ, *'legal'*); замежнай→замежны (ADJ, *'foreign'*)
del 1, + -я	4k	Карэі→Карэя (PROPN, *'Korea'*); дэлегацыі→дэлегацыя (NOUN, *'delegation'*)
del 1, + -ць	4k	прэзентаваў→прэзентаваць (VERB, *'to present'*); гуляў→гуляць (VERB, *'to play'*)
del 1, + -ь	2.5k	Беларусі→Беларусь (PROPN, *'Belarus'*); магчымасці→магчымасць (NOUN, *'opportunity'*)
del 2, + -і	2k	беларускую→беларускі (ADJ, *'Belarusian'*); якая→які (PRON, *'which'*)
del 4, + -цца	1.8k	зацікавілася→зацікавіцца (VERB, *'to become interested'*); адбыліся→адбыцца (VERB, *'to take place'*)

Table 3: Examples of suffix-replacement lemmatization rules induced from the UD Belarusian-HSE training split. Each rule removes a fixed number of characters from the end of the surface form and appends a suffix.

Metric	T1	T2	T3
	UPOS		
Accuracy	98.3±0.1	98.3±0.1	98.2±0.1
Macro-F1	90.2±0.2	90.9±2.1	90.4±1.8
.. w/o INTJ	95.9±0.2	95.8±0.2	95.3±0.3
	FEATS		
Exact bundle acc.	93.4±0.3	**94.5±0.1**	**94.5±0.1**
Macro-F1	69.4±0.9	69.7±0.7	**70.4±0.7**
FV micro-F1	96.7±0.1	**97.2±0.1**	**97.2±0.1**
	LEMMA		
Lemma acc.	90.9±0.2	**92.6±0.2**	**92.5±0.1**
Changed-form acc.	82.7±0.3	**86.3±0.4**	**86.2±0.2**

Table 4: Morphology-aware evaluation on the UD Belarusian-HSE test split. For UPOS, the third row excludes INTJ, which occurs only once in the test set. For FEATS, exact bundle accuracy and macro-F1 are computed for flat full-bundle classification, while feature-value micro-F1 (FV micro-F1) is computed after decomposing predicted and gold bundles into feature-value pairs. Values are mean±std over 15 seeds.

5.3 BelarusianGLUE

To test whether the effects observed in the morphology-aware evaluation transfer to broader natural language understanding, we evaluate the pretrained models on BelarusianGLUE (Aparovich et al., 2025). BelarusianGLUE is a benchmark for Belarusian NLU that contains five expert-crafted datasets, all formulated as binary classification tasks: sentiment analysis, linguistic acceptability, word in context, Winograd schema challenge, and textual entailment. We use all BelarusianGLUE tasks in their standard train/dev/test splits.

For each task and tokenizer, we fine-tune the corresponding pretrained model with a sequence classification head for 10 epochs. We use the same tokenizer as in pretraining, a maximum sequence length of 512, learning rate 2×10^{-5}, weight decay 0.01, warmup ratio 0.06, per-device batch sizes of 16 for training and 32 for evaluation, gradient accumulation of 1, and bfloat16 precision. Each experiment is repeated with 15 random seeds, and we report mean and standard deviation over seeds. Since all tasks are balanced binary classification problems, an accuracy close to 50% corresponds to a trivial baseline.

Table 6 shows a more mixed picture than the morphology-aware evaluation. T2 obtains the largest gain on out-of-domain BelaCoLA (+6.0 macro-F1 over T1) and has the highest mean score on BeRTE-WD and BeSLS, although the latter gains are small, BeRTE-WD remains near the 50% baseline, and the BeSLS differences are small relative to seed variability. In contrast, T1 remains strongest on BeWiC.

Compared with HPLT v2.0 BERT for Belarusian (HPLT-BERT-Be) (Burchell et al., 2025), the three models trained in our controlled setup show mixed results. The clearest advantage is observed on $\text{BelaCoLA}_{\text{out}}$, where T2 reaches 78.9 accuracy, outperforming HPLT-BERT-Be by 21.3 points. Smaller gains over the baseline are observed on BeRTE-WD. In contrast, HPLT BERT Be remains especially stronger on BeSLS, where it outperforms our best model by 7.2 points.

It is important not to overinterpret some of the BelarusianGLUE results. For several tasks, notably BeWSC, BeRTE-WD, and in-domain BelaCoLA, model performance hovers near the 50% majority-class baseline across all tokenization strategies, as well as for the HPLT-BERT-Be.

Taken together with the fact that human performance on BelarusianGLUE is substantially higher than the random baseline, this suggests that the

Feature group	Features	Support	T1	T2	T3
Nominal core	`Animacy, Case, Degree, Gender, Number`	31,778	97.1±0.1	97.5±0.1	97.5±0.1
Verbal core	`Aspect, Mood, Person, Tense, VerbForm, Voice`	7,169	96.3±0.2	97.1±0.2	97.2±0.2
Pronominal/quant.	`NumType, Poss, PronType, Reflex`	1,074	95.9±0.7	95.9±0.4	95.9±0.2
Lexical residual	`Abbr, ExtPos, Foreign, InflClass, NameType, Polarity, Typo, Variant`	1,459	91.5±0.3	90.9±0.4	91.5±0.3

Table 5: Grouped fine-grained FEATS evaluation on the UD Belarusian-HSE test split. Scores are feature-level F1, averaged over 15 seeds.

Task	T1		T2		T3		HPLT-BERT-Be
	F1	Acc.	F1	Acc.	F1	Acc.	Acc.
$\text{BelaCoLA}_{\text{in}}$	41.9±3.7	44.8±3.2	40.6±3.4	43.7±3.8	38.5±4.3	42.5±3.7	48.0
$\text{BelaCoLA}_{\text{out}}$	72.7±9.5	73.5±8.8	**78.7±6.9**	**78.9±6.7**	73.4±7.8	74.1±7.3	57.6
BeRTE-WD	43.1±6.9	50.5±1.1	**51.1±1.8**	**51.9±1.4**	48.8±3.9	50.8±1.2	50.6
BeSLS	83.6±1.1	83.6±1.1	84.0±1.6	84.0±1.6	83.9±1.3	83.9±1.3	91.2
BeWiC	**58.7±1.8**	**59.3±1.7**	56.4±1.9	57.9±1.6	57.0±3.0	57.8±2.3	57.5
BeWSC	48.3±2.3	50.1±0.7	45.2±4.9	49.9±0.6	45.9±4.7	49.8±0.7	49.5

Table 6: BelarusianGLUE test results. Each cell reports macro-F1 or accuracy, in percent, as mean±std over 15 fine-tuning seeds. HPLT BERT Be stands for BERT trained on HPLT dataset (Samuel et al., 2023); results on BelarusianGLUE obtained by Aparovich et al. (2025).

training splits of these specific tasks might be too small or challenging to provide a reliable signal for fine-tuning. Consequently, the lack of significant variance between tokenizers on these datasets should be viewed as a limitation of the benchmark's current capacity to differentiate models, rather than conclusive evidence against morphology-aware tokenization.

6 Conclusion

We presented a controlled study of morphology-aware tokenization for Belarusian masked language models. To isolate the effect of tokenization, we pretrained three RoBERTa-family models under matched conditions, varying only the tokenizer: standard WordPiece, WordPiece trained on morpheme-segmented text, and a dictionary-based hybrid morpheme tokenizer with WordPiece fallback.

Our results show that morphology-aware tokenization is not a universal replacement for standard subword segmentation, but a targeted inductive bias. Such tokenizers yield consistent gains on explicitly morphology-sensitive tasks, especially fine-grained morphological feature prediction and lemmatization. The strongest improvements are observed for changed-form lemmatization and for core nominal and verbal features, indicating that morpheme-aware tokenization helps when the model must recover information encoded in inflectional structure.

The BelarusianGLUE results are more mixed: morphology-aware tokenization improves some tasks but remains comparable to or weaker than standard WordPiece on others. This suggests that general sentence-level NLU benchmarks alone are insufficient for evaluating morphology-aware tokenizers, since they may obscure benefits that become visible only in token-level tasks targeting grammatical structure.

Our comparison also yields a practical takeaway regarding tokenizer design. We observe that the morpheme-trained WordPiece (T2) performs comparably to, and occasionally better than, the more complex hybrid dictionary tokenizer (T3).

Overall, our findings support a selective view of morphology-aware tokenization. Our results are broadly consistent with previous findings for other languages. At the same time, unlike some previous studies (García et al., 2025; Držík and Kapusta, 2026), we cannot claim broad improvements across all tasks. This may be due to differences in tokenization strategy, the properties of the pretraining dataset, or the properties of BelarusianGLUE, whose training splits contain relatively few examples, and should be further explored.

Instead, morpheme-informed tokenization appears most useful for tasks that require access to word-internal structure. Future work should test whether this pattern holds across other Slavic and morphologically rich low-resource languages, and whether simpler morpheme-trained tokenizers such as morpheme-trained WordPiece remain competitive with more complex hybrid designs.

Limitations

Our study has several limitations. First, all experiments are conducted for a single language, Belarusian. This choice makes it possible to perform a controlled analysis in a setting where morphology-aware tokenization is especially relevant, but it also limits the generality of our conclusions. Belarusian is a low-resource East Slavic language with rich inflectional and derivational morphology, and the availability of HPLT, a morphemic resource, and BelarusianGLUE makes it a suitable test case for this study. However, the same patterns may not hold for other Slavic languages, for languages with different morphological typology, or for languages with weaker morphemic resources.

Second, our comparison is restricted to three tokenization strategies: standard WordPiece, WordPiece trained on morpheme-segmented text, and a dictionary-based hybrid tokenizer. While this design isolates the effect of explicit morphological pre-segmentation, it does not cover the full space of possible tokenizers. In particular, we do not compare against Unigram LM tokenization (Kudo, 2018), which has been shown to align more closely with morphological boundaries out-of-the-box compared to frequency-merged methods like BPE or WordPiece (Bostrom and Durrett, 2020). Expanding this controlled setup to include Unigram LM, byte-level models, and recent morphology-constrained subword learning variants remains an important direction for future work.

Third, the quality of the morphology-aware tokenizers depends on the quality and coverage of the underlying segmentation resources. Slounik-WF extends the original dictionary to 332k wordform segmentations and was manually validated, but it is still based on generated paradigms rather than exhaustive manual annotation. For corpus word forms not covered by the dictionary, we rely on automatic segmentation. Although the segmentation model performs strongly, errors in this step may affect tokenizer training and the interpretation of downstream results.

Fourth, our pretraining data come from the Belarusian subset of HPLT v2.0. While HPLT provides large-scale data, it is web-derived and therefore inevitably contains domain imbalance, noise, and duplicated or near-duplicated fragments. We deduplicate exact repeated lines, but this does not remove all boilerplate, near duplicates, or possible language-identification errors. The observed effects of tokenization may therefore partly depend on the properties of this particular pretraining corpus.

Fifth, our downstream evaluation is limited by the available Belarusian benchmarks. BelarusianGLUE provides the most comprehensive broad NLU evaluation currently available for Belarusian, but several of its training splits are small. This makes fine-tuning results more variable and may obscure small but real differences between tokenizers. Conversely, our morphology-sensitive evaluation is based on Universal Dependencies tasks, which are well suited for testing morphosyntactic generalization but do not cover all possible uses of morphological information, such as generation, machine translation, question answering, or robustness to noisy input.

Sixth, the inherently higher token fertility of morphology-aware tokenizers reduces the effective word-level context size for a fixed token window, which may disadvantage them on downstream tasks requiring long-range dependencies.

Finally, our models are RoBERTa-family masked language models trained under matched conditions. This controlled setup is appropriate for isolating tokenization effects, but it does not establish whether the same conclusions would hold true for larger models, decoder-only architectures, encoder–decoder models, longer pretraining schedules, or different masking strategies. Future work should test whether the selective benefits observed here persist across model scales, objectives, and languages.

References

Maksim Aparovich, Volha Harytskaya, Vladislav Poritski, Oksana Volchek, and Pavel Smrz. 2025. BelarusianGLUE: Towards a natural language understanding benchmark for Belarusian. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 511–527, Vienna, Austria. Association for Computational Linguistics.

Mikhail Arkhipov, Maria Trofimova, Yuri Kuratov, and Alexey Sorokin. 2019. Tuning multilingual transformers for language-specific named entity recognition. In *Proceedings of the 7th Workshop on Balto-Slavic Natural Language Processing*, pages 89–93, Florence, Italy. Association for Computational Linguistics.

Kaj Bostrom and Greg Durrett. 2020. Byte pair encoding is suboptimal for language model pretraining. In

Findings of the Association for Computational Linguistics: EMNLP 2020, pages 4617–4624, Online. Association for Computational Linguistics.

Laurie Burchell, Ona de Gibert, Nikolay Arefyev, Mikko Aulamo, Marta Bañón, Pinzhen Chen, Mariia Fedorova, Liane Guillou, Barry Haddow, Jan Hajič, Jindřich Helcl, Erik Henriksson, Mateusz Klimaszewski, Ville Komulainen, Andrey Kutuzov, Joona Kytöniemi, Veronika Laippala, Petter Mæhlum, Bhavitvya Malik, and 16 others. 2025. An expanded massive multilingual dataset for high-performance language technologies (HPLT). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 17452–17485, Vienna, Austria. 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.

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.

Dávid Držík and Jozef Kapusta. 2026. The importance of morphology-aware subword tokenization for nlp tasks in slovak language modeling. *Expert Systems with Applications*, 312:131492.

Alba Táboas García, Piotr Przybyła, and Leo Wanner. 2025. Exploring morphology-aware tokenization: A case study on Spanish language modeling. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 30505–30518, Suzhou, China. Association for Computational Linguistics.

Rob van der Goot, Anette Jensen, Emil Allerslev Schledermann, Mikkel Wildner Kildeberg, Nicolaj Larsen, Mike Zhang, and Elisa Bassignana. 2025. MorSeD: Morphological segmentation of Danish and its effect on language modeling. In *Proceedings of the Joint 25th Nordic Conference on Computational Linguistics and 11th Baltic Conference on Human Language Technologies (NoDaLiDa/Baltic-HLT 2025)*, pages 223–229, Tallinn, Estonia. University of Tartu Library.

Haris Jabbar. 2024. Morphpiece : A linguistic tokenizer for large language models. *Preprint*, arXiv:2307.07262.

Mikkel Wildner Kildeberg, Emil Allerslev Schledermann, Nicolaj Larsen, and Rob van der Goot. 2025. From smør-re-brød to subwords: Training llms on danish, one morpheme at a time. *Preprint*, arXiv:2504.01540.

Taku Kudo. 2018. Subword regularization: Improving neural network translation models with multiple subword candidates. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 66–75, Melbourne, Australia. Association for Computational Linguistics.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

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. *Preprint*, arXiv:1907.11692.

L. S. Mormysh, A. M. Bordovich, and L. M. Shakun. 2005. *School morpheme dictionary of the Belarusian language [SHkol'ny marfemny slovnik belaruskaj movy]*. Aversev, Minsk.

Dmitry Morozov, Lizaveta Astapenka, Anna Glazkova, Timur Garipov, and Olga Lyashevskaya. 2025. BERT-like models for Slavic morpheme segmentation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6795–6815, Vienna, Austria. Association for Computational Linguistics.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Jan Hajič, Christopher D. Manning, Sampo Pyysalo, Sebastian Schuster, Francis Tyers, and Daniel Zeman. 2020. Universal Dependencies v2: An evergrowing multilingual treebank collection. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4034–4043, Marseille, France. European Language Resources Association.

Michal Olbrich and Zdeněk Žabokrtský. 2026. Morphological segmentation with neural networks: Performance effects of architecture, data size, and cross-lingual transfer in seven languages. In *Text, Speech, and Dialogue*, pages 275–286, Cham. Springer Nature Switzerland.

Hyunji Hayley Park, Katherine J. Zhang, Coleman Haley, Kenneth Steimel, Han Liu, and Lane Schwartz. 2021. Morphology matters: A multilingual language modeling analysis. *Transactions of the Association for Computational Linguistics*, 9:261–276.

Kyubyong Park, Joohong Lee, Seongbo Jang, and Dawoon Jung. 2020. An empirical study of tokenization

strategies for various Korean NLP tasks. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 133–142, Suzhou, China. Association for Computational Linguistics.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, and 2 others. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

David Samuel, Andrey Kutuzov, Lilja Øvrelid, and Erik Velldal. 2023. Trained on 100 million words and still in shape: BERT meets British National Corpus. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 1954–1974, Dubrovnik, Croatia. Association for Computational Linguistics.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2020. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *Preprint*, arXiv:1910.01108.

Svetlana O Savchuk, Timofey Arkhangelskiy, Anastasiya A Bonch-Osmolovskaya, Ol'ga V Donina, Yuliya N Kuznetsova, Ol'ga N Lyashevskaya, Boris V Orekhov, and Mariya V Podryadchikova. 2024. Russian National Corpus 2.0: New opportunities and development prospects. *Voprosy Jazykoznanija*, 2024(2):7–34.

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

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.

Clara Vania and Adam Lopez. 2017. From characters to words to in between: Do we capture morphology? In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2016–2027, Vancouver, Canada. Association for Computational Linguistics.

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, and 3 others. 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.

Dmitry Zmitrovich, Aleksandr Abramov, Andrey Kalmykov, Vitaly Kadulin, Maria Tikhonova, Ekaterina Taktasheva, Danil Astafurov, Mark Baushenko, Artem Snegirev, Tatiana Shavrina, Sergei S. Markov, Vladislav Mikhailov, and Alena Fenogenova. 2024. A family of pretrained transformer language models for Russian. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 507–524, Torino, Italia. ELRA and ICCL.

A Risks

The risks associated with this work are limited but not absent. We train encoder-only masked language models for Belarusian and evaluate them on classification and morphology-oriented tasks; the models are not designed for open-ended text generation. Therefore, the direct risk of generating harmful, misleading, or abusive content is lower than for decoder-only generative models. However, the pretrained encoders may still be used as components in downstream systems, including content moderation, user profiling, information retrieval, or text classification pipelines. Errors or biases in such systems could have negative effects if deployed in sensitive domains.

A second risk comes from the use of large-scale web-derived pretraining data. Although HPLT provides a valuable resource for low-resource language modeling, web corpora may contain toxic content, stereotypes, political bias, duplicated boilerplate, or mislabeled language data. These properties can be inherited by pretrained models and may affect downstream predictions. Models trained on web data may overrepresent some registers, orthographic practices, domains, or ideological contexts while underrepresenting others.

A third risk concerns the morphemic resources used to construct morphology-aware tokenizers. Morpheme segmentation reflects particular annotation decisions and may encode a relatively standardized or dictionary-based view of the language. As a result, morphology-aware tokenization may privilege standard written Belarusian over dialectal, mixed, colloquial, or non-standard forms. This could reduce robustness for underrepresented varieties and may unintentionally reinforce a narrow view of linguistic correctness.

Finally, our results could be overinterpreted as showing that morphology-aware tokenization is universally better for Belarusian or for morphologically rich languages in general. Our findings are more specific: morphology-aware tokenization improves explicitly morphological tasks but does not uniformly improve broad NLU tasks. We therefore caution against deploying morphology-aware tokenizers as a default choice without validating them on the target task, domain, and user population.

B Scientific Artifacts

Our work uses external datasets, pre-trained models, and software libraries. We also create several new artifacts, including an expanded morphemic resource, tokenizers, pretrained language models, and the code used for tokenizer construction, pretraining, and evaluation. We plan to release the newly created artifacts after blind review. All scientific artifacts we used are permitted for scientific use, provided they comply with the license requirements. The artifacts we create will be published under licenses that do not violate the licenses of the source materials and models.

B.1 Existing Resources

Pretraining data. For masked language model pretraining, we use the Belarusian subset of HPLT v2[5] (Burchell et al., 2025). HPLT is a large-scale web-derived multilingual corpus available under the Creative Commons CC0 license. We use it as an existing resource and do not collect or annotate new pretraining data.

We used the cleaned Belarusian subset of HPLT v2. HPLT v2 documents are annotated for personally identifiable information using a multilingual PII detection tool, which records character offsets of detected PII in the metadata. The cleaned HPLT release also applies quality and compliance filtering, including robots.txt compliance and filtering of URLs from an adult-content blacklist; the HPLT authors additionally report manual inspection for pornographic and unnatural content. We did not attempt to deanonymize any individuals and did not release raw HPLT text as part of our artifacts.

Morphemic resource. We use **Slounik**, a Belarusian morpheme segmentation resource derived from the School Morpheme Dictionary of the Belarusian Language (Mormysh et al., 2005; Morozov et al., 2025). Slounik provides surface morpheme segmentations and morpheme-type labels, and serves as the starting point for constructing the expanded word-form-level resource used in this work. The dataset is available under Creative Commons CC-BY-NC-SA 4.0 license.

Evaluation benchmarks. For downstream evaluation, we use existing Belarusian resources and benchmarks. We use the Belarusian-HSE treebank[6] from Universal Dependencies (Nivre et al., 2020) for UPOS tagging, morphological feature prediction, and lemmatization. The treebank is available under CC BY-SA 4.0 license. We also use BelarusianGLUE[7] (Aparovich et al., 2025), which is available under Apache license 2.0, for broad NLU evaluation. These datasets are used as released; we do not create additional annotations for them.

Pre-trained models used for segmentation experiments. In the morpheme segmentation stage, we use existing BERT-like encoders. We utilized eight pre-trained models from HuggingFace[8]:

1. `roberta-small-belarusian`[9]. This model is monolingual (Belarusian), has 16M parameters, and is available under the CC-BY-SA 4.0 license.

2. `hplt_bert_base_2_0_bel-Cyrl`[10] (Burchell et al., 2025). This model is monolingual (Belarusian), has 124M parameters and is available under the Apache 2.0 license.

3. `ruBert-base`[11]. (Zmitrovich et al., 2024). This model is monolingual (Russian), has 178M parameters, and is available under the MIT license.

4. `ruRoberta-large`[12] (Zmitrovich et al., 2024). This model is monolingual (Russian), has 355M parameters, and is available under the MIT license.

[5] https://hplt-project.org/datasets/v2.0

[6] https://github.com/UniversalDependencies/UD_Belarusian-HSE

[7] https://huggingface.co/datasets/maaxap/BelarusianGLUE

[8] https://huggingface.co/

[9] https://huggingface.co/KoichiYasuoka/roberta-small-belarusian

[10] https://huggingface.co/HPLT/hplt_bert_base_2_0_bel-Cyrl

[11] https://huggingface.co/ai-forever/ruBert-base

[12] https://huggingface.co/ai-forever/ruRoberta-large

5. `SlavicBERT`[13] (Arkhipov et al., 2019). This model is multilingual (Bulgarian, Czech, Polish, and Russian), has 180M parameters, and is available under the Apache 2.0 license.

6. `bert-base-multilingual-cased`[14] (Devlin et al., 2019). This model was trained on 104 languages, has 177M parameters, and is available under Apache 2.0 license.

7. `distilbert-base-multilingual-cased`[15] (Sanh et al., 2020). This model was trained on 104 languages, has 134M parameters, and is available under Apache 2.0 license.

8. `XLM-Roberta-large`[16] (Conneau et al., 2020). This model was trained on 100 languages, has 550M parameters, and is available under MIT license.

Software Libraries. We use PyTorch (Paszke et al., 2019) 2.6.0 for neural model training and Transformers (Wolf et al., 2020) 5.8.0 for implementing and fine-tuning BERT/RoBERTa-family models. We use the Hugging Face Tokenizers[17] 0.22.2 for WordPiece tokenizer training and serialization. Additional data processing and evaluation scripts are implemented in Python using standard scientific and NLP libraries.

B.2 Created Artifacts

Slounik-WF. We create **Slounik-WF**, an expanded word-form-level version of Slounik. Slounik-WF extends the original lemma-level morphemic resource with generated and validated inflected word forms. It can be used to train and evaluate morpheme segmentation models and to construct morphology-aware tokenizers for Belarusian. We are planning to publish the Slounik-WF dataset under Creative Commons CC-BY-NC-SA 4.0 license after blind review.

Tokenizers. We create three tokenizers for the controlled pretraining experiments:

1. **T1**: a standard WordPiece tokenizer trained on the Belarusian HPLT corpus;

2. **T2**: a WordPiece tokenizer trained on text pre-segmented with morphemic information;

3. **T3**: a dictionary-based hybrid morpheme tokenizer that applies Slounik-WF segmentations when available and falls back to WordPiece otherwise.

These tokenizers will be released together with the configuration files needed to reproduce their training and application under Creative Commons CC-BY-NC-SA 4.0 license after blind review.

Pretrained language models. We train three RoBERTa-family masked language models that differ only in their tokenizer. The models are pretrained on the same Belarusian HPLT corpus under matched conditions and correspond to the T1, T2, and T3 tokenization settings. The resulting model checkpoints will be released under Apache 2.0 license after blind review.

Code. We implement and release our code[18] used for 1) training and applying the three tokenizers; 2) preparing the pretraining corpus; 3) pretraining the masked language models; 4) fine-tuning and evaluating the models on BelarusianGLUE and Universal Dependencies. The code is implemented primarily in Python.

C Computational Experiments

We pretrained our models on a single multi-GPU node with 10 NVIDIA A100 GPUs using distributed data parallel training (approximately 10 multi-GPU hours in total), and fine-tuned them on three NVIDIA A100 GPU nodes (approximately 100 single-GPU hours in total).

D Human Annotation

No external human annotators were recruited for this study. The only human annotation effort specific to this work concerns the preparation and validation of Slounik-WF, the expanded Belarusian word-form-level morphemic resource used to construct the morphology-aware tokenizers. This work was performed by a member of the author team with relevant linguistic expertise. Since no external annotators were hired, no separate compensation scheme was used.

[13] https://huggingface.co/DeepPavlov/bert-base-bg-cs-pl-ru-cased
[14] https://huggingface.co/google-bert/bert-base-multilingual-cased
[15] https://huggingface.co/distilbert/distilbert-base-multilingual-cased
[16] https://huggingface.co/FacebookAI/xlm-roberta-large
[17] https://github.com/huggingface/tokenizers
[18] https://anonymous.4open.science/r/belamorph-5BB9

The evaluation benchmarks used in the paper were adopted from existing resources. We use Universal Dependencies Belarusian-HSE and BelarusianGLUE as released, without collecting additional annotations. The pretraining corpus was also taken from an existing source, HPLT v2.0, and no human annotators were involved in its collection, labeling, or manual filtering for the purposes of this study.

E AI Assistant

Although we did not use AI assistants writing the original text of the article, we used an AI assistant as a translator and proof-reader. We also used an AI assistant to design codebase architecture and to refactor parts of the codebase for improved readability. Some of the services used for searching scientific literature and datasets also rely on AI.

F Slounik-WF Construction

The expansion procedure proceeds from lemma-level segmentations. For each lemma, we derive a stem by removing the inflectional ending and, for verbs, the infinitive suffix *-ць*. We then compare the stem with the generated word form. If the word form can be obtained through right-side concatenation, the added string is segmented according to POS-specific ending and suffix patterns. If the difference involves internal variation, the change is checked against a predefined inventory of morphophonemic alternations, including root and suffix alternations. The resulting segmentations were then subjected to two rounds of selective manual validation by a native Belarusian speaker with linguistic training. The validation was used to refine the alternation lists and POS-specific patterns and to remove systematic errors. The remaining known source of ambiguity concerns rare cases of morphological homonymy in verb forms. In our validation, such cases were rare.

Table 7 summarizes the resulting resource. *Slounik-WF* contains 332,497 unique word-form segmentations, including 150,543 verbal forms, 86,442 noun forms, 93,037 adjective forms, and 2,475 adverbs. The dataset contains 8,186 unique morphemes, and the average segmentation consists of 3.86 morphemes. We use this resource both as training data for segmentation models and as a high-precision source of dictionary segmentations in the hybrid tokenizer described below.

Characteristic	*Slounik*	*Slounik-WF*
Entries / segmentations	35,219	332,497
Nouns	13102	86,442
Adjectives	7682	93,037
Verbs	12242	150,543
Adverbs	2,475	2,475
Unique morphemes	6,350	8,186
Unique roots	6,095	6,095
Average morphemes per word	3.77	3.86

Table 7: Brief characteristics of *Slounik* and *Slounik-WF*.

Parameter	Value
Layers	12
Hidden size	768
Attention heads	12
Intermediate size	3072
Vocabulary size	50,000
Maximum sequence length	512
Training steps	100,000
Optimizer	AdamW
Peak learning rate	1×10^{-4}
LR schedule	Linear with warmup
Warmup ratio / steps	0.06 / 6,000
Weight decay	0.01
Adam β_1, β_2	0.9, 0.98
Adam ϵ	1×10^{-6}
Per-device train batch size	32
Per-device eval batch size	32
Gradient accumulation steps	1
Precision	bfloat16

Table 8: Pretraining hyperparameters shared by all three models.

G Segmentation Setup

For the CNN, we used a 10-layer character-level ResNet with a hidden size of 256, a filter size of 3, and a dropout rate of 0.1. For the LSTM, we utilized a two-layer character-level BiLSTM with a hidden dimension of 256 and a dropout rate of 0.1. Both models were trained for 30 epochs using the cross-entropy loss function. All BERT-like models were fine-tuned for 30 epochs with a learning rate of 4×10^{-6}.

H Pretraining Setup

The hyperparameters listed in Table 8 were used for pretraining all models.

Figure 1 shows the MLM loss during pretraining. The upper panel reports the absolute evaluation loss for the three pretrained models, while the lower panel shows pairwise loss differences between tokenizers. This representation is useful because the T2 and T3 curves nearly overlap in absolute loss: both morphology-aware tokenizers follow a very similar pretraining trajectory, whereas T1 remains separated due to its different, coarser

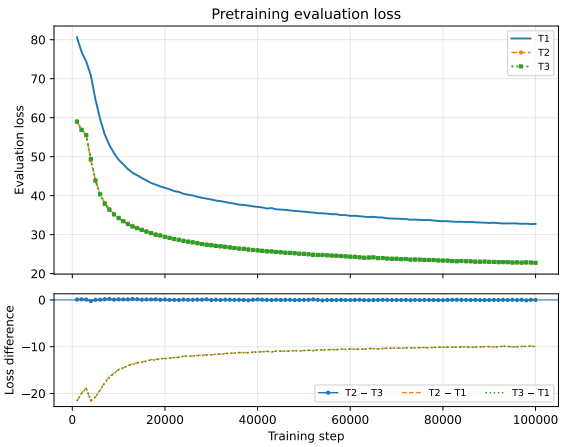

Figure 1: Evaluation loss during MLM pretraining.

WordPiece prediction units.

I UD Evaluation - Detailed Results

I.1 UPOS tagging

The per-tag results, provided in Table 9, show that the differences between tokenizers are generally small for high-frequency UPOS classes, but they are not uniformly distributed. T2 obtains the best score for several content-word categories, including `NOUN`, `ADJ`, `VERB`, `PROPN`, `PRON`, and `DET`. The gains are modest in absolute terms, typically between 0.1 and 0.6 percentage points.

At the same time, T1 remains strongest for several function-word categories, including `ADP`, `CCONJ`, `ADV`, `NUM`, `PART`, `SCONJ`, and `AUX`. This suggests that morphology-aware tokenization does not provide a uniform advantage for UPOS tagging, especially for short, closed-class items whose category is often recoverable from the whole token or local context.

Overall, T2 provides the most consistent improvements for morphologically informative open-class categories, while T1 is more stable for closed-class and low-morphology categories; T3 does not show a clear advantage over the other two tokenizers.

I.2 Feats tagging

The per-feature results, provided in Table 10, show a clearer advantage for morphology-aware tokenization than the UPOS results. T2 and T3 outperform T1 on most high-frequency morphological features, including `Number`, `Gender`, `Animacy`, `Degree`, `VerbForm`, `Voice`, `Aspect`, `Tense`, `Mood`, `Person`, and `PronType`. The strongest gains are observed for features that are directly expressed by inflectional morphology: `Gender` improves by +1.8 points with T2 and +1.7 with T3, `Voice` by about +1.1, and verbal categories such as `Aspect`, `Tense`, and `Mood` also show consistent improvements. This supports the main hypothesis that morpheme-aware tokenization is especially useful when the model must recover grammatical information encoded in word-internal structure.

At the same time, the advantage is not uniform across all features. T1 remains best for `Case` and for several low-support or more idiosyncratic features, such as `NumType`, `Foreign`, `Abbr`, `Poss`, `Polarity`, `Variant`, `Reflex`, and `InflClass`. Many of these categories have relatively small support, so their differences should be interpreted cautiously. Comparing the two morphology-aware tokenizers, T2 is strongest for several broad nominal and adjectival features, such as `Number`, `Gender`, `Degree`, `VerbForm`, `Voice`, and `PronType`, whereas T3 performs best on `Animacy`, `Aspect`, `Tense`, `Mood`, `NameType`, and `Person`. Overall, both morphology-aware tokenizers improve feature prediction over standard WordPiece for the main inflectional categories, with T2 appearing slightly more stable and T3 showing stronger gains on several verbal and pronominal features.

I.3 Lemmatization

To better characterize the behavior of the T1- and T2-based lemmatizers, we compared the lemmatization outputs obtained for one of the random seeds. Out of 15,997 evaluated tokens, both models predict the correct lemma in 14,267 cases (89.19%), while both fail in 981 cases (6.13%). Importantly, asymmetric cases are not balanced: T2 corrects 509 errors made by T1 (3.18%), whereas T1 corrects only 240 errors made by T2 (1.50%). Thus, the morphology-aware tokenizer yields a net gain of 269 token-level corrections over the standard WordPiece tokenizer. This suggests that the improvement is not driven by a small number of isolated examples, but reflects a systematic shift in the model's behavior on a non-negligible subset of inflected forms.

The list below provides a qualitative comparison of lemmatization errors corrected by the morphology-aware model. The selected examples are cases where T1 predicts an incorrect lemma, whereas T2 returns the gold lemma. A common property of these examples is that the correct pre-

UPOS	Support	T1	T2	T3	T2–T1	T3–T1	T3–T2	Best
NOUN	4623	98.9±0.2	99.0±0.1	98.9±0.1	+0.1	+0.0	-0.1	T2
PUNCT	2754	99.7±0.1	99.7±0.0	99.8±0.0	+0.0	+0.1	+0.1	T3
ADJ	1893	97.8±0.2	97.9±0.3	97.7±0.2	+0.1	-0.1	-0.2	T2
ADP	1643	99.9±0.0	99.2±0.1	99.2±0.1	-0.6	-0.7	-0.0	T1
VERB	1366	98.5±0.2	98.7±0.3	98.7±0.1	+0.2	+0.1	-0.0	T2
PROPN	946	96.1±0.7	96.2±0.3	95.9±0.3	+0.1	-0.2	-0.3	T2
CCONJ	641	98.6±0.2	98.6±0.2	98.5±0.2	-0.0	-0.1	-0.1	T1
ADV	397	94.6±0.4	94.0±0.4	93.8±0.5	-0.5	-0.8	-0.3	T1
X	395	96.3±0.4	96.2±0.3	96.8±0.3	-0.1	+0.5	+0.6	T3
PRON	366	97.3±0.4	97.5±0.5	97.2±0.6	+0.2	-0.1	-0.3	T2
DET	350	96.4±0.9	97.0±0.7	96.8±0.7	+0.6	+0.4	-0.2	T2
NUM	233	95.1±1.8	94.2±1.1	93.8±0.6	-0.8	-1.2	-0.4	T1
PART	134	92.4±0.6	91.4±0.9	91.1±0.8	-1.0	-1.4	-0.4	T1
SCONJ	95	92.8±1.1	92.6±1.5	92.7±1.6	-0.2	-0.1	+0.0	T1
AUX	90	96.9±1.1	96.4±1.1	96.5±0.8	-0.4	-0.4	+0.1	T1
SYM	70	83.0±1.6	83.7±1.6	77.9±1.8	+0.7	-5.1	-5.8	T2
INTJ	1	0.0±0.0	13.3±35.2	11.1±30.0	+13.3	+11.1	-2.2	T2

Table 9: UPOS tagging accuracy by part of speech on the UD Belarusian-HSE test split. Values are mean±std over seeds. Deltas are absolute percentage-point differences.

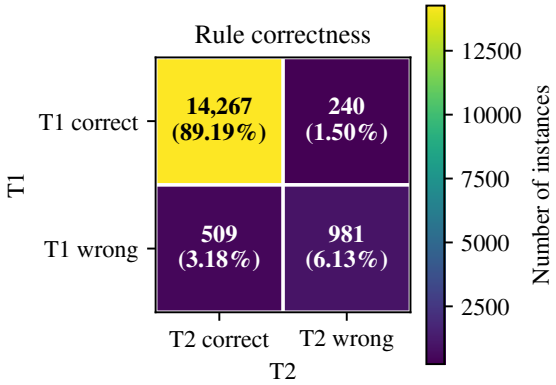

Figure 2: Instance-level comparison of T1 and T2 lemmatization rule correctness on the UD Belarusian-HSE test split. Rows indicate T1 correctness and columns indicate T2 correctness.

diction requires modifying the inflectional ending while preserving the lexical base. Standard WordPiece often represents the relevant word form as a single token or isolates only a small final fragment, whereas T2 produces a segmentation closer to the morphemic structure of the word.

Across the examples, T2 tends to recover the lemma through a linguistically plausible transformation: it separates the inflectional ending and replaces it with the ending expected in the citation form. T1 errors, by contrast, often look like stem-preserving but morphologically ill-formed substitutions. Although these examples do not by themselves establish a causal mechanism, they are consistent with the quantitative results: morphology-aware tokenization provides the model with a more transparent representation of inflectional structure, which is particularly useful for lemmatization.

1. ажыццяўляюць → ажыццяўляць (VERB)
   - Context: Органы і падраздзяленні па надзвычайных сітуацыях [...] **ажыццяўляюць** дзейнасць у галіне папярэджання і ліквідацыі...
   - Translation: Emergency agencies and units of the Republic of Belarus [...] **carry out** activities in the field of prevention and elimination...
   - T1 prediction: ажыццяўлаваць, rule -4+аваць
   - T2 prediction: ажыццяўляць, rule -3+ць
   - T1 tokenization: ажыццяўляюць
   - T2 tokenization: а + жы + цц + яўл + я + юць
2. патрабаванняў → патрабаванне (NOUN)
   - Context: Дзяржаўны пажарны нагляд за захаваннем [...] **патрабаванняў** заканадаўства аб пажарнай бяспецы...
   - Translation: State fire supervision over compliance with [...] the **requirements** of fire safety legislation...
   - T1 prediction: патрабаваннь, rule -2+ь
   - T2 prediction: патрабаванне, rule -2+е
   - T1 tokenization: патрабаванняў
   - T2 tokenization: па + траб + ав + а + нн + яў

Feature	Support	T1	T2	T3	T2–T1	T3–T1	T3–T2	Best
Number	8903	98.5±0.1	98.6±0.1	98.6±0.1	+0.1	+0.1	-0.0	T2
Case	7846	96.1±0.2	96.0±0.1	96.0±0.2	-0.1	-0.1	+0.1	T1
Gender	7213	95.8±0.4	97.6±0.1	97.6±0.1	+1.8	+1.7	-0.1	T2
Animacy	5752	97.6±0.1	98.0±0.1	98.0±0.1	+0.4	+0.4	+0.0	T3
Degree	2064	97.1±0.2	97.3±0.1	97.1±0.1	+0.2	+0.0	-0.2	T2
VerbForm	1427	97.0±0.3	97.6±0.2	97.6±0.2	+0.7	+0.6	-0.1	T2
Voice	1426	97.2±0.2	98.3±0.1	98.3±0.2	+1.1	+1.1	-0.1	T2
Aspect	1370	92.7±0.3	93.5±0.3	93.8±0.3	+0.8	+1.1	+0.3	T3
Tense	1180	96.3±0.3	97.2±0.3	97.3±0.2	+0.9	+1.0	+0.1	T3
Mood	1019	98.1±0.2	98.6±0.2	98.7±0.1	+0.5	+0.6	+0.1	T3
NameType	944	90.4±0.4	90.3±0.3	91.0±0.4	-0.1	+0.6	+0.7	T3
Person	747	97.3±0.3	97.9±0.3	98.1±0.3	+0.6	+0.7	+0.1	T3
PronType	707	96.9±0.3	97.4±0.2	97.3±0.2	+0.5	+0.5	-0.0	T2
NumType	210	91.7±0.6	90.6±1.6	90.4±0.8	-1.1	-1.3	-0.2	T1
Foreign	171	93.8±1.1	93.0±1.2	93.3±1.3	-0.8	-0.4	+0.4	T1
Abbr	163	98.0±0.6	97.1±0.7	97.0±0.5	-0.9	-1.0	-0.1	T1
Poss	121	97.8±0.4	97.2±0.8	97.7±0.6	-0.6	-0.1	+0.5	T1
Polarity	81	100.0±0.0	97.8±0.9	97.1±0.8	-2.2	-2.9	-0.6	T1
Variant	62	83.6±1.9	81.7±1.5	83.3±1.5	-1.9	-0.3	+1.6	T1
Reflex	36	96.0±0.0	95.2±1.0	94.8±1.1	-0.8	-1.2	-0.4	T1
InflClass	29	79.7±4.1	76.9±2.6	79.3±4.1	-2.9	-0.4	+2.4	T1
Typo	6	1.9±7.4	0.0±0.0	0.0±0.0	-1.9	-1.9	+0.0	T1
ExtPos	3	10.0±20.7	6.7±17.6	0.0±0.0	-3.3	-10.0	-6.7	T1

Table 10: FEATS prediction F1 by feature on the UD Belarusian-HSE test split. Values are mean±std over seeds. Deltas are absolute percentage-point differences.

3. памяшканнямі → памяшканне (NOUN)
   - Context: ... забеспячэнне, сацыяльна-бытавое абслугоўванне і забеспячэнне жылымі **памяшканнямі** работнікаў органаў і падраздзяленняў па надзвычайных...
   - Translation: ... provision, social and everyday services, and provision of residential **premises** for employees of bodies and units for emergency...
   - T1 prediction: памяшканак, rule `-4+ак`
   - T2 prediction: памяшканне, rule `-3+е`
   - T1 tokenization: памяшкання + ##мі
   - T2 tokenization: па + мяш + к + а + нн + ямі

4. сканцэнтравана → сканцэнтраваць (VERB)
   - Context: Вакол Пушкінскай плошчы **сканцэнтравана** шмат сілавікоў.
   - Translation: Many security officers are **concentrated** around Pushkinskaya Square.
   - T1 prediction: сканцэнтравыць, rule `-3+ыць`
   - T2 prediction: сканцэнтраваць, rule `-2+ць`
   - T1 tokenization: сканцэнтравана
   - T2 tokenization: с + канцэнтр + ав + а + н + а

5. тэлебачаньні → тэлебачаньне (NOUN)
   - Context: ... ўпотайкі глядзеў прамую трансьляцыю па польскім **тэлебачаньні**...
   - Translation: ... secretly watched the live broadcast on Polish **television**...
   - T1 prediction: тэлебачаньт, rule `-2+т`
   - T2 prediction: тэлебачаньне, rule `-1+е`
   - T1 tokenization: тэлебачаньні
   - T2 tokenization: тэл + е + бач + а + ньн + і

6. вытворчасці → вытворчасць (NOUN)
   - Context: ... форма гаспадарча-працоўнай дзейнасці грамадзян па **вытворчасці** сельскагаспадарчай прадукцыі, заснаваная на выкарыстанні зямельных...
   - Translation: ... a form of economic and labor activity of citizens involving the **production** of agricultural products, based on the use of land...
   - T1 prediction: вытворчасца, rule `-1+а`
   - T2 prediction: вытворчасць, rule `-1+ь`
   - T1 tokenization: вытворчас + ##ці

- T2 tokenization: вы + твор + ч + асц + і

7. перапрацоўцы → перапрацоўка (NOUN)
    - Context: Дзейнасць грамадзян [...] па вытворчасці, **перапрацоўцы** і рэалізацыі выробленай імі сельскагаспадарчай прадукцыі...
    - Translation: The activities of citizens [...] involving the production, **processing**, and sale of agricultural products produced by them...
    - T1 prediction: перапрацоўца, rule -1+а
    - T2 prediction: перапрацоўка, rule -2+ка
    - T1 tokenization: перапрацоўцы
    - T2 tokenization: пера + прац + оў + ц + ы

To further examine the qualitative differences between T1 and T2, we compared word length and tokenizer granularity across correct and incorrect lemmatization predictions. The results suggest that T1 errors are associated with longer word forms: words incorrectly lemmatized by T1 are substantially longer on average than those it lemmatizes correctly (8.87 vs. 5.38 characters). This tendency is even stronger in the subset where T1 is wrong but T2 is correct, where the average word length reaches 9.22 characters.

The contrast is also visible in tokenization granularity. In the cases where T1 is wrong and T2 is correct, 56.4% of the forms are represented by T1 as a single token, while 95.9% are split by T2 into multiple units. This supports the interpretation that morpheme-informed tokenization is particularly helpful for longer morphologically complex forms, where a standard WordPiece tokenizer may provide little internal structure.