TamilTok: Morphologically-Informed Tokenization for Tamil

Surendhar Muthukumar and Aaricia Herygers and Lisa Beinborn
Institute of Computer Science, University of Goettingen, Germany
surendhar.m@stud.uni-goettingen.de, firstname.lastname@uni-goettingen.de

Abstract

Tokenization is fundamental to neural language modeling, yet for Tamil it remains largely adapted from general-purpose multilingual models without systematic consideration of the rich agglutinative morphology. We introduce TamilMorph, a large-scale dataset of more than 480,000 morphologically segmented Tamil word forms. Building on this new resource, we develop TamilTok, a morphology-aware tokenization framework that incorporates explicit morpheme structure into tokenizer training. We benchmark Tamil tokenization quality across multiple tokenization algorithms and vocabulary configurations and find that our approach improves both morphological alignment and downstream performance compared to previous approaches. Our morphological resource for Tamil and our systematic empirical analyses can guide future developments of tokenization for morphologically rich languages.

1 Introduction

Tokenization shapes the interface between raw symbolic text and computational representations in large language models. The choices of the tokenization algorithm and the granularity of the vocabulary determine how textual input is segmented into machine-learnable units, affecting model efficiency, preservation of semantic information, and generalization capabilities (Petrov et al., 2023). Subword tokenization has been optimized for the rather shallow morphological structure of English, and it remains an open research question how well the frequency-driven segmentation approach captures more complex agglutinative processes (Ahia et al., 2023; Arnett and Bergen, 2025).

Tamil is a Dravidian language spoken by more than 80 million people in India, Sri Lanka, Singapore, Malaysia, and communities of the global diaspora. It is written in an abugida script that encodes morphophonemic processes known as sandhi phenomena: systematic sound changes when two morphemes are combined. Tamil morphology is agglutinative, which means that suffixes can be stacked and the combined meaning can be derived systematically from the meaning of parts due to direct form–function correspondences. For example, the word விருந்தினர்களுக்கும் ("for the guests as well") is an agglutinative form derived from the root விருந்தினர் ("guest"). It is formed by sequentially adding the plural marker -கள் and the dative marker -உக்கும் (which additionally encodes the meaning "also/too").

Tamil uses systematic suffixation to inflect nouns across multiple grammatical cases and encodes tense, aspect, and agreement of verbs via sequential morpheme addition. Due to the high morphological regularity, a single lexical root can generate dozens of valid surface forms. The productive stacking of suffixes presents a challenge for frequency-driven subword tokenization, which often fragments morphemes into frequent surface forms, leading to disrupted morphological boundaries, especially in multilingual models (Bayram et al., 2025; Chintha and Konduru, 2025; Jabbar, 2023; Parra, 2024).

Despite Tamil's linguistic richness and large speaker base, dedicated studies on Tamil tokenization are scarce (Krishnan and Ragavan, 2021; T K et al., 2024). In practice, Tamil is usually processed with multilingual tokenizers, rather than dedicated monolingual ones. The design assumptions of multilingual models are largely shaped by high-resource analytic languages such as English, which are structured by whitespaces and exhibit only limited inflection. As a consequence, a disproportionate portion of shared multilingual vocabularies is allocated to high-frequency units from resource-rich languages. This imbalance is further compounded by the distinct Tamil script, re-

Proceedings of the Sixth Workshop on Speech, Vision, and Language Technologies for Dravidian Languages, pages 52–61

sulting in minimal orthographic overlap with other languages. As illustrated in Figure 1, morphologically rich languages such as Tamil are often oversegmented, causing longer sequences with reduced morphological coherence leading to higher computational cost (Ahia et al., 2023; Petrov et al., 2023).

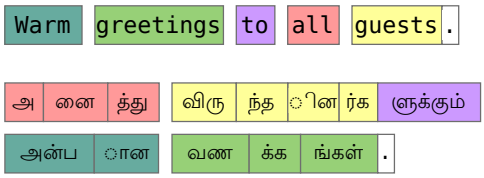

Figure 1: Example of over-fragmentation of a Tamil sentence by the Gemma tokenizer (Gemma Team, 2025). Words (visualized as boxes) are split into multiple subword tokens and token splits (expressed by |) do not align with morphological boundaries. For the English translation, the same tokenizer keeps all words as a single token.

Recent work demonstrates that linguistically-informed segmentation can improve morphological alignment and downstream performance in morphologically rich languages. Approaches integrating morphological supervision (Pan et al., 2020), rule-based constraints (Asgari et al., 2025), morphological analyzers (Matthews et al., 2018), architecture-level morphology modeling (Kumar M et al., 2010; Ataman and Federico, 2018), or morphology-aware pre-tokenization (Brahma et al., 2025) have shown gains in preserving morpheme boundaries and reducing over-segmentation. Yet, progress for Tamil remains limited, largely due to lacking large-scale high-quality morphological segmentation resources for tokenizer training and evaluation.

We introduce `TamilMorph`, a large-scale morphological segmentation dataset that contains over 480,000 valid morphological word forms with annotated morpheme boundaries to address the lack of high-quality resources for Tamil. Based on this gold standard, we present our morphology-aware tokenization framework `TamilTok`. By decomposing surface forms into valid morpheme sequences prior to subword learning, we improve morphological coherence and downstream performance on named entity recognition while maintaining competitive compression efficiency.

Our dataset and tokenizer are publicly available to support further research on Tamil NLP[1].

[1] https://gitlab.gwdg.de/huds/projects/tamiltok

2 Related Work

Tokenization is commonly conducted using frequency-driven subword segmentation methods, but their suitability for morphologically rich languages has been questioned. While prior work has demonstrated the potential benefits of linguistically-informed segmentation, progress remains uneven due to limited large-scale, task-aligned morphological datasets for many languages.

2.1 Subword Tokenization

Subword tokenization splits text input into sequences of frequently co-occurring characters. In contrast to word-level segmentation, it can capture novel sequences that were not seen during training (by fragmenting them into smaller parts) while maintaining a manageable vocabulary size. The most widely adopted approaches include Byte Pair Encoding (BPE) (Sennrich et al., 2016), which greedily merges the most frequent adjacent symbols; WordPiece (Devlin et al., 2019), which incrementally builds a subword vocabulary using likelihood-based merges that improve corpus modeling; Unigram Language Modeling (UnigramLM) (Kudo, 2018), which formulates tokenization as a probabilistic segmentation problem and iteratively prunes subword units that minimally contribute to the likelihood of the model.

Frameworks such as SentencePiece (Kudo and Richardson, 2018) enable language-independent preprocessing on raw text without relying on whitespaces, while byte-level models such as ByT5 (Xue et al., 2022) operate directly on raw bytes. Despite architectural differences, all subword segmentation approaches are fundamentally frequency-driven and largely language-agnostic without considering typological variation in morphological structure, causing large cross-lingual differences in tokenization quality.

2.2 Tokenization of Morphologically Complex Languages

Tokenizing morphologically rich languages presents unique challenges that standard subword approaches often fail to address adequately. Tamil, in particular, has been identified as one of the most fragmented languages in multilingual tokenization regimes, causing increased computational cost and reduced semantic coherence (Ahia et al., 2023; Petrov et al., 2023).

To address these issues, several studies explore morphology-aware tokenization strategies. For example, Pan et al. (2020) incorporate morphological preprocessing into BPE for neural machine translation for Turkish and Uyghur, Ataman et al. (2019) propose hierarchical character-based decoding that explicitly preserves morpheme boundaries during translation, and Matthews et al. (2018) introduce morpheme-level embeddings.

More recent analyses examine the influence of improved morphological alignment on downstream task performance. Previous research finds positive correlations between morphology-aware representations and downstream task accuracies (Vemula et al., 2025), but others conclude that the relationship is task- and language-dependent (Arnett et al., 2025), or resolved by equalizing data quantity (Arnett and Bergen, 2025). However, their evaluation does not include Tamil or other Dravidian languages. In contrast, Brahma et al. (2025) find that morphology-driven pre-tokenization improves performance for Hindi and Marathi using neural segmentation combined with corpus-level heuristics. These results indicate that morphology-aware tokenization can yield tangible benefits when properly aligned with the linguistic structure.

3 The TamilMorph Dataset

Previous approaches for morphological segmentation of Tamil morphology use finite-state analyzers (Sarveswaran et al., 2021), rule-based segmentation (Krishnan and Ragavan, 2021), or unsupervised approaches (Virpioja et al., 2013), but the existing systems only capture a very small set of morphological phenomena and do not generalize robustly. In order to assess morphological boundaries on a larger scale while maintaining grammatical accuracy, we introduce `TamilMorph`: an extensive dataset specifically designed for identifying linguistically reliable morpheme boundaries. It provides broad coverage of morphological phenomena such as inflectional and derivational patterns of nouns and verbs (including phonologically-conditioned alternations) while adhering to grammatical constraints.

3.1 Dataset Construction

Figure 2 visualizes our hybrid construction strategy that combines linguistic rules with controlled computational generation, ensuring both linguistic validity and a broad morphological coverage. The process follows a reverse engineering approach: Tamil words are first synthetically generated according to derivation and inflection rules while keeping track of segmentation boundaries. Using this controlled procedure for manipulating root forms, we avoid relying on an imprecise morphological analyzer. Instead, we focus on extracting high-quality root forms and linguistically valid rules.

Root Extraction We extract root morphemes directly from the annotated Tamil corpus provided by the Tamil Virtual Academy (Tamil Virtual Academy, 2022). Since our framework requires explicit identification of base forms, only the root information associated with each POS-tagged word form is utilized. After normalization and deduplication of the 320,448 annotated word forms, we obtained 28,209 noun roots and 3,078 verb roots.

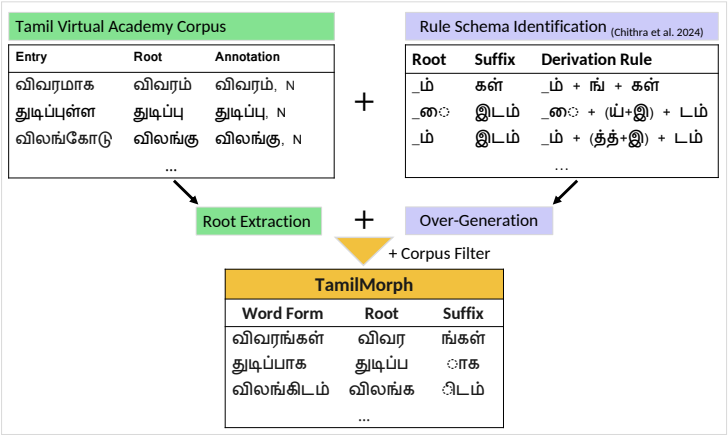

Figure 2: The dataset construction process for `TamilMorph`. We extract root morphemes and derivation rules from linguistic resources, over-generate all possible forms, and then filter based on corpus frequencies.

Rule Schema Identification We derive morphological rules from a grammar reference book for Tamil (Chithra et al., 2024), which documents the canonical patterns of well-formed derivation in Tamil, including sandhi processes governing phonological alternations at morpheme boundaries. As illustrated in Figure 2 in the *Formation Rule* column, the locative suffix இடம் surfaces differently depending on the phonological properties of the root ending ம் and ை. Distinct sandhi operations, indicated within parentheses, account for these variations, but the morphological function of the suffix remains the same.

Morphological Derivation Using the extracted root morphemes and linguistic rules, we implement a synthetic expansion procedure to overgenerate inflected and derived forms. For each root, all permissible suffix chains are applied according to morphological rules and phonological constraints. This process captures inflectional patterns for number, case, tense, mood, aspect, and voice for both nouns and verbs. In a second stage, the over-generated surface forms are checked against a strict corpus-based filter: only forms that occur at least once in the IndicNLP corpus (Kunchukuttan et al., 2020) are kept to avoid overly artificial generations. The dataset statistics are provided in Table 1.

Stage	Type	Count
Generation	Verbal suffixes	31
	Nominal suffixes	34
	Unique roots	31,287
	Generated forms	53,217,115
Filtered	**Word Forms**	**481,099**
	Unique root forms	101,821
	Unique suffixes	1,037
	Avg. suffix length	3.8

Table 1: Dataset statistics for `TamilMorph`.

3.2 TamilMorph

Using the procedure described above, we created `TamilMorph` to extract Tamil word forms with explicitly annotated morpheme boundaries that correspond to the correct linguistic structure. Each entry consists of a surface-form word decomposed into two components: a root and a corresponding suffix. To maintain structural consistency, all non-root morphemes associated with a word are combined into a single suffix unit, resulting in a fixed two-part representation (root + suffix) for every word form.[2] An example of this representation is shown in Figure 2.

The dataset systematically captures a broad spectrum of agglutinative inflectional and derivational Tamil patterns. Beyond serving as a linguistic resource, `TamilMorph` serves as the backbone of our `TamilTok` framework. The annotated segmentations enable supervised training of a neural morphological segmentation model, which generalizes beyond the dataset vocabulary and functions as an upstream component in the morphology-aware tokenization pipeline. Furthermore, the explicit morpheme annotations and broad lexical coverage provide a gold standard for evaluating tokenizer behavior using morphological alignment metrics such as MorphScore F_1 (Arnett and Bergen, 2025) and comparing tokenization strategies.

4 Experimental Setup

In our analysis, we compare the Tamil tokenization quality of newly trained monolingual tokenizers to a range of pretrained multilingual tokenizers: mBERT (Devlin et al., 2019), XLM-R (Conneau et al., 2020), mT5 (Xue et al., 2021), Gemma (Gemma Team, 2025), mBART (Liu et al., 2020), IndicBERT (Doddapaneni et al., 2023). Our standardized evaluation setup compares the influence of the training corpus, the tokenization algorithm, and the vocabulary size.

4.1 Tokenizer Training

We train monolingual Tamil tokenizers using standardized normalization and preprocessing pipelines; the implementation details can be found in our code repository provided in the supplementary material. We use the three subword tokenization algorithms BPE (Sennrich et al., 2016), WordPiece (Devlin et al., 2019), and UnigramLM (Kudo, 2018), as implemented in the SentencePiece package.[3]

We use the Tamil portion of IndicNLP (Doddapaneni et al., 2023) with 68,237,343 lines and Tamil Wikipedia (Wikimedia Foundation, 2023) with 4,716,963 lines as training data. After filtering all lines containing non-Tamil characters, we retain 10,693,052 lines from Indic NLP and 2,997,493 from Wikipedia and ap-

[2] We found that the coarse-grained distinction provided more robust information for the tokenizer in pilot experiments, but we also provide the more fine-grained boundaries in our resource repository.

[3] `pypi.org/project/sentencepiece/0.2.0/`

ply Unicode normalization.[4] We vary the vocabulary size k of each tokenizer ($k \in \{3000, 5000, 8000, 16000, 32000\}$). Small values of k are comparable to the limited vocabulary allocation of Tamil in a shared multilingual vocabulary; larger values are more common for monolingual tokenizers.

4.2 Intrinsic Evaluation

We evaluate intrinsic tokenization quality along two complementary dimensions: information compression and linguistic plausibility. Information compression focuses on the efficiency of information reduction achieved by the tokenizer. In addition to the compression rate (ratio of characters to tokens), we analyze word-level fertility (average number of subword tokens per word) and Rényi efficiency (Zouhar et al., 2023), which measures how efficiently the tokenizer balances the token distribution over the vocabulary. Higher values for Rényi efficiency indicate that the usage of the vocabulary is closer to a uniform distribution. While this suggests more efficient resource allocation, it is counterintuitive from a linguistic perspective because in all natural languages, vocabulary usage follows a Zipfian distribution with a long tail of rare words. A highly efficient tokenizer does not necessarily use morphologically plausible boundaries to segment the input. Using our `TamilMorph` dataset, we can quantify the linguistic plausibility of the tokenizer by comparing subword splits to morpheme boundaries. We quantify the difference between the splits proposed by the tokenizer and the morphological boundaries between root and suffix using the Creutz–Linden F_1 (Morph-F_1) calculation (Creutz and Linden, 2004). Unlike the default MorphScore implementation (Arnett et al., 2025), we compute boundary statistics within a unified evaluation framework to accommodate format differences and allow consistent per-tokenizer aggregation. The resulting F_1-score penalizes both over- and under-segmentation, providing a direct and interpretable measure of how well a tokenizer preserves morphologically meaningful structure.

4.3 Downstream Performance

In order to quantify the interaction between intrinsic tokenization quality and downstream performance, we train a `TamilBERT` model using a standard encoder-only transformer architecture consisting of eight transformer blocks with eight attention heads. The model is pre-trained with a masked-language-modeling objective on the IndicNLP corpus using the different tokenizers and the smallest vocabulary size ($k = 3000$). The pre-trained TamilBERT models are finetuned on the WikiAnn Tamil NER dataset (Pan et al., 2017), which provides BIO-tagged annotations for the categories person (PER), location (LOC), and organization (ORG). The dataset contains 38,728 instances that are split into train (50%), dev (25%) and test (25%) sets. Finetuning is performed with a sequence length of 128 tokens, a batch size of 32, and a learning rate of 5×10^{-5} for 10 epochs.

5 TamilTok: Tokenization with Morphological Information

Subword tokenizers typically involve a pre-tokenization step that splits the input based on whitespaces. However, standard whitespace tokenization fails to capture internal morphological boundaries in agglutinative languages like Tamil. We aim at introducing a pre-tokenization step that splits the input based on morphological boundaries before learning subwords. Using our new dataset `TamilMorph`, we can now train a morphological segmentation model capable of decomposing unseen surface forms into valid morpheme sequences. We model segmentation as a conditional sequence generation task using ByT5 (Xue et al., 2022). We employ a byte-level architecture for this segmentation task as it robustly handles out-of-vocabulary characters without inducing prior biases, allowing the model to learn morphological boundaries directly from the provided data. We fine-tune the ByT5-Small configuration (300M parameters) on 503,292 root-suffix pairs in `TamilMorph` using a 90/10 train–test split without providing any auxiliary linguistic features or lexicon constraints. The model is trained for 5 epochs with a learning rate of 5×10^{-5} and deterministic decoding. The segmentation algorithm correctly identifies 96% of the boundaries between the root and the suffix on the test set.

If we apply our novel morphological segmentation algorithm as a pre-tokenization step, we force the subword algorithm to perform merge operations within morphologically valid units to reduce cross-morpheme fragmentation. The segmentation algorithm alone cannot be used directly as in-

[4] Unicode normalization ensures that visually similar Tamil characters are mapped to the same code points.

Lang	Type	Config	Fert.↓	Comp.	Rényi	Morph-F_1	NER-F_1
English	Multilingual	mBERT	1.32	4.61	.60	-	-
Tamil	Multilingual	mBERT	3.64	2.55	.47	.24	-
		XLM-R	2.47	3.79	.54	.31	-
		mT5	2.54	3.65	.55	.29	-
		Gemma	2.39	3.92	.56	.20	-
		mBART	2.47	3.79	.54	.31	-
		IndicBERT	3.39	2.73	.37	.11	-
	Monolingual	Byte-level BPE	2.51	3.73	**.87**	.26	.64
		Unigram	**2.27**	**3.96**	.85	.33	.65
		WordPiece	2.59	3.60	.84	.26	.65
	Morph. informed	`TamilTok`	2.50	3.75	.83	**.35**	**.67**

Table 2: Comparison of pre-trained multilingual tokenizers to our small monolingual tokenizers ($k = 3000$) and to our `TamilTok` variant with morphologically-informed pre-tokenization for the intrinsic evaluation metrics fertility, compression, and Rényi efficiency, for the alignment with morphological boundaries (Morph-F_1) and for downstream performance on named entity recognition (NER-F_1). For comparison, we additionally report the results for English using the mBERT tokenizer.

put for a neural model as it does not use a fixed vocabulary size and does not necessarily align with the surface structure of the input.

6 Results

In Table 2, we compare the tokenization quality of Tamil when using existing pretrained multilingual tokenizers to newly trained monolingual tokenizers.

6.1 Tokenizer Evaluation

For a fair comparison, we first provide the results for the monolingual tokenizers with the smallest vocabulary size ($k = 3000$). Fertility, compression ratio and Rényi efficiency are measured using parallel Tamil-English sentences from the Flores-101 dataset (Goyal et al., 2022), whereas the Morph-F_1 and NER-F_1 are measured using the `TamilMorph` test set and WikiAnn (Pan et al., 2017), respectively. The results confirm our anecdotal impression that existing pre-trained multilingual tokenizers perform poorly on Tamil inputs: high fertility and low compression rate indicate over-fragmentation that violates morphological boundaries. The mBERT model tokenizes English words into 1.32 subword tokens on average and Tamil words into 3.64 subword tokens. The monolingual tokenizers yield better information compression (especially the one based on the UnigramLM algorithm), and the high values for Rényi efficiency indicate a better allocation of the vocabulary. Our `TamilTok` shows a substantial improvement in morphological alignment without reducing information compression and downstream performance.

The choice of the training corpus exhibits only a marginal effect on alignment quality. For the UnigramLM algorithm, the tokenizers trained on the IndicNLP corpus achieve slightly higher fertility (3.91) and Morph-F_1 (0.33) compared to those trained on Wikipedia (3.87; 0.32). However, the differences are minimal, indicating that both corpora produce broadly comparable subword segmentation quality when trained under controlled and balanced conditions.

6.2 Tokenization Parameters

For the following experiments, we chose the IndicNLP corpus due to its larger overall token count and higher number of unique word forms to better examine the influence of the vocabulary size.

Tokenization Algorithm Figure 3 illustrates the influence of the vocabulary size and the tokenization algorithm on the F_1 and fertility metrics. Across all configurations, using UnigramLM consistently leads to better morphological alignment than BPE or WordPiece. Crucially, this advantage is not accompanied by substantially higher fertility, indicating that the gains cannot be attributed to increased subword fragmentation. Rather, the results suggest that the probabilistic objective underlying UnigramLM more effectively captures morphologically coherent boundaries than deterministic merge-based approaches.

Vocabulary Size Increasing the vocabulary systematically reduces fertility, reflecting improved compression efficiency through longer and more

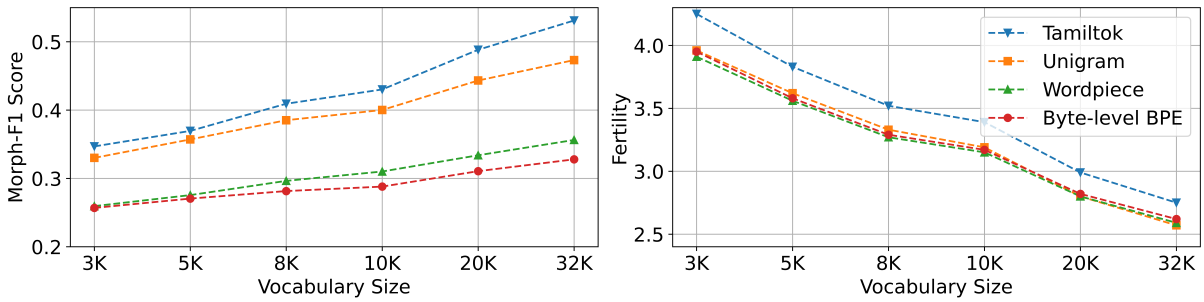

Figure 3: Effect of tokenization algorithm and vocabulary size on morphological alignment (Morph-F_1, left) and fertility (right). Both metrics improve with increasing vocabulary size and the UnigramLM algorithm consistently outperforms BPE and WordPiece in preserving morphological boundaries.

consolidated subword units. At the same time, morphological alignment increases steadily across the entire vocabulary range. The absence of saturation within the examined range indicates that morphological segmentation quality generally benefits strongly from significantly larger vocabulary sizes independent of the segmentation algorithm.

Evaluating fertility jointly with alignment therefore provides a more faithful characterization of tokenizer quality and clearly establishes the superiority of UnigramLM. Furthermore, as expected, larger vocabulary sizes offer increased capacity for extracting linguistically meaningful subword units, contributing to improved alignment alongside reduced fragmentation.

7 Discussion

We observe three main observations in our experiments. First, multilingual tokenizers exhibit substantial over-segmentation for Tamil, as reflected in high fertility and low morphological alignment scores. This confirms prior concerns (Ahia et al., 2023; Arnett and Bergen, 2025; Petrov et al., 2023) that frequency-driven shared vocabularies disproportionately disadvantage morphologically rich and non-Latin-script languages. For Tamil, the resulting fragmentation disrupts morpheme boundaries and increases sequence length.

Second, the tokenization algorithm and the vocabulary size have a stronger influence on morphological alignment than the choice of the training corpus. UnigramLM consistently outperforms BPE and WordPiece in Morph-F_1 across controlled settings and both compression and alignment benetfit from larger vocabulary sizes. Importantly, fertility alone does not sufficiently capture qualitative tokenization differences, underscoring the need for linguistically motivated evaluation.

Third, explicitly combining morphological information with existing frequency-based tokenization algorithms in `TamilTok` further improves morphological alignment, in line with previous work (Brahma et al., 2025; Bayram et al., 2025; Jabbar, 2023; Krishnan and Ragavan, 2021). Although morphology-aware preprocessing introduces a slight increase in fertility, the gain in Morph-F_1 indicates better preservation of linguistically meaningful units. This suggests that minor compression trade-offs may be justified when structural coherence is improved.

The proposed `TamilMorph` dataset plays a central role in enabling these analyses. Its rule-derived boundary-level annotations provide a consistent gold standard for evaluating tokenizer–morpheme agreement and support supervised training of neural segmentation models. It can be used for training and benchmarking in future research on Tamil morphological segmentation and tokenization.

8 Conclusion

Our work introduces two contributions to morphology-aware NLP for Tamil. We present `TamilMorph`, a large-scale morphological segmentation dataset comprising more than 480,000 linguistically validated word forms with explicit morpheme segmentations. The dataset offers a reproducible gold standard for evaluating morphological alignment and supports supervised morphological segmentation.

The proposed `TamilTok` algorithm, a morphology-aware tokenization framework, integrates neural morphological segmentation into subword training. By injecting morpheme structure prior to tokenizer training, the framework improves morphological alignment while maintaining competitive compression efficiency

and downstream performance.

Our approach can be extended to other agglutinative languages, provided suitable morphological resources are available. Overall, our work demonstrates that linguistically-informed tokenization leads to more structurally coherent representations and offers a practical pathway toward improved modeling of morphologically rich languages.

Limitations

We extracted the root forms from the Tamil Virtual Academy corpus (Tamil Virtual Academy, 2022) without systematically ensuring their lexical minimality. Manual examination indicates that some entries labeled as roots in the corpus correspond to compound forms (that function as surface-level bases) rather than irreducible morphological units. We do not expect this limitation to affect the segmentation performance or structural consistency of the dataset, but it may influence the strict linguistic validity of the root annotations and thus, to an extent, affect the representation quality of linguistic stems on downstream tasks.

The derivational rules used for generating the forms in the dataset are not exhaustive. The full range of possible derivational suffixes cannot be derived from a single corpus or limited word-formation references. Expanding the inventory of suffixes will require broader linguistic investigation and the integration of additional lexical resources.

Our corpus-based filter for ensuring lexical validity is overly strict. It might exclude forms that occur with high frequency in domains not covered in the IndicNLP corpus.

Acknowledgments

This research is partially supported by the zukunft.niedersachsen program of the VolkswagenStiftung (LB) and by a VENI grant (Vl.Veni.211C.039) from the Nederlandse Organisatie voor Wetenschappelijk Onderzoek (AH, LB).

References

Orevaoghene Ahia, Sachin Kumar, Hila Gonen, Jungo Kasai, David Mortensen, Noah Smith, and Yulia Tsvetkov. 2023. Do all languages cost the same? tokenization in the era of commercial language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9904–9923, Singapore. Association for Computational Linguistics.

Catherine Arnett and Benjamin Bergen. 2025. Why do language models perform worse for morphologically complex languages? In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 6607–6623. Association for Computational Linguistics.

Catherine Arnett, Marisa Hudspeth, and Brendan O'Connor. 2025. Evaluating morphological alignment of tokenizers in 70 languages. In *Proceedings of the ICML 2025 Tokenization Workshop (TokShop)*.

Ehsaneddin Asgari, Yassine El Kheir, and Mohammad Ali Sadraei Javaheri. 2025. MorphBPE: A morpho-aware tokenizer bridging linguistic complexity for efficient LLM training across morphologies. *Preprint*, arXiv:2502.00894.

Duygu Ataman and Marcello Federico. 2018. Compositional representation of morphologically-rich input for neural machine translation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 305–311, Melbourne, Australia. Association for Computational Linguistics.

Duygu Ataman, Orhan Firat, Mattia A. Di Gangi, Marcello Federico, and Alexandra Birch. 2019. On the importance of word boundaries in character-level neural machine translation. In *Proceedings of the 3rd Workshop on Neural Generation and Translation*, pages 187–193, Hong Kong. Association for Computational Linguistics.

M Ali Bayram, Ali Arda Fincan, Ahmet Semih Gümüş, Sercan Karakaş, Banu Diri, Savaş Yıldırım, and Demircan Çelik. 2025. Tokens with meaning: A hybrid tokenization approach for nlp. *arXiv preprint arXiv:2508.14292*.

Maharaj Brahma, NJ Karthika, Atul Singh, Devaraj Adiga, Smruti Bhate, Ganesh Ramakrishnan, Rohit Saluja, and Maunendra Sankar Desarkar. 2025. Morphtok: Morphologically grounded tokenization for Indian languages. *arXiv e-prints*, pages arXiv–2504.

Deepthi Chintha and Nanda Vikas Konduru. 2025. Survey of tokenization mechanisms in multilingual large language models with a focus on Indian languages. *Journal of Emerging Technologies and Innovative Research*.

V Chithra, V Balamurugan, N Rajendiran, M Tamilarasan, and P Kavitha. 2024. Tamizhai pizhaiyindri ezhudhivom. https://www.tamilvu.org/sites/default/files/notice/Thamizhai_Pizhaiyinri_Ezhuthuvom.pdf.

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.

Mathias Johan Philip Creutz and Bo Krister Johan Linden. 2004. Morpheme segmentation gold standards for Finnish and English. In *Publications in Computer and Information Science: Report A77*. Helsinki University of Technology.

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.

Sumanth Doddapaneni, Rahul Aralikatte, Gowtham Ramesh, Shreya Goyal, Mitesh M. Khapra, Anoop Kunchukuttan, and Pratyush Kumar. 2023. Towards leaving no Indic language behind: Building monolingual corpora, benchmark and models for Indic languages. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12402–12426, Toronto, Canada.

Gemma Team. 2025. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*.

Naman Goyal, Cynthia Gao, Vishrav Chaudhary, Peng-Jen Chen, Guillaume Wenzek, Da Ju, Sanjana Krishnan, Marc'Aurelio Ranzato, Francisco Guzmán, and Angela Fan. 2022. The Flores-101 evaluation benchmark for low-resource and multilingual machine translation. *Transactions of the Association for Computational Linguistics*, 10:522–538.

Haris Jabbar. 2023. Morphpiece: A linguistic tokenizer for large language models. *arXiv preprint arXiv:2307.07262*.

Arjun Sai Krishnan and Seyoon Ragavan. 2021. Morphology-aware meta-embeddings for Tamil. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Student Research Workshop*, pages 94–111, Online. Association for Computational Linguistics.

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. Association for Computational Linguistics.

Anand Kumar M, Dhanalakshmi V, Soman K.P, and Rajendran S. 2010. A novel data driven algorithm for Tamil morphological generator. *International Journal of Computer Applications*, 6(12):52–56.

Anoop Kunchukuttan, Divyanshu Kakwani, Satish Golla, Gokul N.C., Avik Bhattacharyya, Mitesh M. Khapra, and Pratyush Kumar. 2020. AI4Bharat-IndicNLP Corpus: Monolingual corpora and word embeddings for Indic languages. *arXiv preprint arXiv:2005.00085*.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Austin Matthews, Graham Neubig, and Chris Dyer. 2018. Using morphological knowledge in open-vocabulary neural language models. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1435–1445.

Xiaoman Pan, Boliang Zhang, Jonathan May, Joel Nothman, Kevin Knight, and Heng Ji. 2017. Cross-lingual name tagging and linking for 282 languages. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1946–1958, Vancouver, Canada. Association for Computational Linguistics.

Yirong Pan, Xiao Li, Yating Yang, and Rui Dong. 2020. Morphological word segmentation on agglutinative languages for neural machine translation. *Preprint*, arXiv:2001.01589.

Iñigo Parra. 2024. Morphological typology in BPE subword productivity and language modeling. In *Latinx in AI@ NeurIPS 2024*.

Aleksandar Petrov, Emanuele La Malfa, Philip Torr, and Adel Bibi. 2023. Language model tokenizers introduce unfairness between languages. In *Advances in Neural Information Processing Systems*, volume 36, pages 36963–36990. Curran Associates, Inc.

Kengatharaiyer Sarveswaran, Gihan Dias, and Miriam Butt. 2021. Thamizhi Morph: A morphological parser for the Tamil language. *Machine Translation*, 35(1):37–70.

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.

Jothir Adithya T K, Nithish Kumar S, Felicia Lilian J, and Mahalakshmi S. 2024. Monolingual text summarization for Indic languages using LLMs. In *Proceedings of the 21st International Conference on Natural Language Processing (ICON)*, pages 94–101, AU-KBC Research Centre, Chennai, India. NLP Association of India (NLPAI).

Tamil Virtual Academy. 2022. Tamil annotated corpus. https://github.com/Tamil-Virtual-Academy/Tamil-Annotated-Corpus. Accessed: 2026-02-25.

Saketh Reddy Vemula, Sandipan Dandapat, Dipti Sharma, and Parameswari Krishnamurthy. 2025. Rethinking tokenization for rich morphology: The dominance of unigram over BPE and morphological alignment. In *The 14th International Joint Conference on Natural Language Processing and The 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics*, pages 232–252, Mumbai, India. Association for Computational Linguistics.

Sami Virpioja, Peter Smit, Stig-Arne Grönroos, and Mikko Kurimo. 2013. Morfessor 2.0: Python implementation and extensions for morfessor baseline. Aalto University publication series SCIENCE + TECHNOLOGY, 25/2013.

Wikimedia Foundation. 2023. Wikimedia downloads. Accessed: 10-10-2025.

Linting Xue, Aditya Barua, Noah Constant, Rami Al-Rfou, Sharan Narang, Mihir Kale, Adam Roberts, and Colin Raffel. 2022. ByT5: Towards a token-free future with pre-trained byte-to-byte models. *Transactions of the Association for Computational Linguistics*, 10:291–306.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online. Association for Computational Linguistics.

Vilém Zouhar, Clara Meister, Juan Gastaldi, Li Du, Mrinmaya Sachan, and Ryan Cotterell. 2023. Tokenization and the noiseless channel. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5184–5207, Toronto, Canada. Association for Computational Linguistics.