Byte Pair Encoding Is All You Need For Automatic Bengali Speech Recognition

Ahnaf Mozib Samin[*]
Queen's University, Canada
University of Groningen, The Netherlands
University of Malta, Malta
ahnaf.samin@queensu.ca

Abstract

Byte pair encoding (BPE) emerges as an effective tokenization method for tackling the out-of-vocabulary (OOV) challenge in various natural language and speech processing tasks. Recent research highlights the dependency of BPE subword tokenization's efficacy on the morphological nature of the language, particularly in languages rich in inflectional morphology, where fewer BPE merges suffice for generating highly productive tokens. Motivated by this, our study empirically identifies the optimal number of BPE tokens for Bengali, a language known for its morphological complexity, thus enhancing out-of-distribution automatic speech recognition (ASR) performance. Experimental evaluation reveals that an excessively high number of BPE tokens can lead to overfitting, while approximately 500-1000 tokens result in superior OOV performance. Furthermore, we conduct a comparative analysis of BPE with character-based and unigram-based tokenization methods. By introducing BPE tokenization to Bengali ASR, we achieve a substantial reduction in the word error rate (WER) from 66.44% in our character-based baseline system to 63.80% on the LB-ASRTD eval set and from 46.34% to 42.80% on the SHRUTI eval set, both of which include out-of-distribution data.

1 Introduction

The performance of an automatic speech recognition system is contingent on its core components including acoustic feature extraction, mapping acoustic features to tokens using the Gaussian Mixture Model/Hidden Markov Model (GMM-HMM) or neural networks, and language model-based rescoring of the outputs from connectionist temporal classification (CTC), etc (Graves et al., 2006). As for segmentation, either word-level or subword-level tokens can be modeled to build ASR systems. However, word-level modeling faces a challenge due to the vast number of words in a language, which exceeds the typical vocabulary size of an ASR system. As a result, word-level modeling is susceptible to the out-of-vocabulary (OOV) problem (Livescu et al., 2012). An OOV word refers to a word that was not encountered during model training but appears during the inference phase.

Subword units are known as effective solutions to tackle the OOV issue in natural language processing (NLP) (Sennrich et al., 2016). Examples of subword units include phonemes, characters, unigrams, and byte pair encoding (BPE) tokens, etc (Kudo, 2018; Gage, 1994; Sennrich et al., 2016). Phonemes represent the smallest units of sound, and after training with phonemes, a model can infer new words. Creating a lexicon that maps each word to its corresponding phonemes, however, requires domain expertise as well as a substantial amount of time and effort for manual annotation (Harwath and Glass, 2014). Character-based ASR models are easier to develop since mapping between words and their corresponding characters can be done automatically (Chan et al., 2016). Furthermore, training a model with a limited number of characters in a language is more computationally efficient than training a word-based model. Unigram language modeling is another segmentation technique that removes tokens based on language model perplexity, initially applied in machine translation (Kudo, 2018).

BPE subword tokenization is first utilized in neural machine translation (NMT) and has gained widespread usage due to their ability to handle OOV words effectively (Gage, 1994; Sennrich et al., 2016; Radford et al., 2018). Subsequent studies implement BPE-based subword modeling in the speech processing domain (Synnaeve et al., 2020; Yusuyin et al., 2023). For BPE, the number of merge operations determines the number of generated tokens/subwords. From the work of Gutierrez-

[*] Work performed during the Erasmus Mundus Joint MSc. program at the University of Groningen and the University of Malta.

Proceedings of the Second Workshop on Bangla Language Processing (BLP-2025), pages 1–7
December 23, 2025 ©2025 Association for Computational Linguistics

Vasques et al. (2023) on 47 diverse languages, it has been found that in languages characterized by extensive inflectional morphology, there is a tendency to generate highly productive subwords during the initial merging steps. Conversely, in languages with limited inflectional morphology, idiosyncratic subwords tend to play a more prominent role (Parra, 2024). Therefore, the characteristics of subwords and the required number of merges in BPE tokenization are contingent upon the morphological nature of the respective language. Moreover, an empirical study is conducted by incrementally increasing the BPE merges going from characters to words (Gutierrez-Vasques et al., 2021). The authors reported that around 200 BPE merges result in the most similar distribution across different languages.

Since the optimal number of BPE merges cannot be universally determined for different languages with varied types of morphology (Gutierrez-Vasques et al., 2023), in this study, we empirically determine the number of BPE merges needed for a highly inflectional language—Bengali to achieve superior out-of-distribution ASR performance. Bengali is an Indo-Aryan language spoken in Bangladesh and India and poses challenges to the development of robust ASR systems due to its intricate morphological forms and inadequate research (Ali et al., 2008; Samin et al., 2021). Subsequently, we compare the results of BPE-based tokenization with segmentation approaches based on characters and unigrams by performing a cross-dataset evaluation, aiming to understand their effectiveness for handling out-of-distribution data. To the best of our knowledge, this investigation exploiting different subword modeling approaches is conducted for the first time for Bengali ASR.

The rest of the paper is structured as follows: a comprehensive background study on BPE and unigram language modeling, along with a review of related work in the speech processing domain is provided in Section 2. The methodology of our experiments are described in Section 3. Details about the experiment setup are provided in Section 4. Results are discussed in Section 5. The conclusion and outlines the future directions are provided in Section 6.

2 Background

2.1 Byte pair encoding

Byte pair encoding is a data compression algorithm, which was applied in NMT in 2016 (Gage, 1994; Sennrich et al., 2016).

Algorithm 1 Byte-pair encoding (Gage, 1994; Sennrich et al., 2016; Bostrom and Durrett, 2020)

```
$S \leftarrow$ set of strings (Approx. 40k Bengali words)
$n \leftarrow$ target vocab size
**procedure** BPE($S, n$)
    $V \leftarrow$ all unique characters in $S$
    **while** $|V| < n$ **do**
        **Step 1** Merge tokens $a$ and $b$, where
            $a, b \in V$ and represent the most
            frequent bigram in $S$
        **Step 2** Create a new token $ab$ by
            concatenating $a$ and $b$
        **Step 3** Add $ab$ to $V$
        **Step 4** Replace each bigram occurrence
            of $a, b$ tokens in $S$ with $ab$
    **end while**
    **return** $V$
**end procedure**
```

BPE algorithm takes a set of strings S and aims to create a vocabulary V with a target size of n. It iteratively merges the most frequent bigram in S into a new token, updating the vocabulary and replacing occurrences of the merged tokens in the original strings. The algorithm continues until the vocabulary size reaches the desired target size n and returns the final vocabulary V.

2.2 Unigram language modeling

Unigram language modeling (LM) was first applied in NMT in 2018 and compared the performance to that of BPE (Kudo, 2018). Unigram language modeling algorithm takes a set of strings S and aims to create a vocabulary V with a target size of n. It starts by initializing V with all substrings occurring more than once in S (without crossing words). The algorithm then iteratively prunes the vocabulary by estimating the token 'loss' L_t for each token t in V using the unigram language model θ. The tokens with the highest L_t values are removed from V until its size reaches the target vocabulary size n. Finally, the algorithm fits the final unigram language model θ to S and returns V and θ as the resulting vocabulary and language model, respectively.

Algorithm 2 Unigram LM (Kudo, 2018; Bostrom and Durrett, 2020)

```
$S \leftarrow$ set of strings (Approx. 40k Bengali words)
$n \leftarrow$ target vocab size
procedure UNIGRAM($S, n$)
    $V \leftarrow$ all substrings occurring more than
        once in $S$ (not crossing words)
    while $|V| > n$ do
        Build the unigram language model $\theta$
        with $S$
        for $t$ in $V$ do
            $L_t \leftarrow pplx_{\theta}(S) - pplx_{\theta'}(S)$
            where $\theta^{'}$ is the LM without token t
        end for
        Remove min($|V| - n$, $\lfloor \alpha |V| \rfloor$) of the
        tokens $t$ with highest $L_t$ from $V$ ,
        where $\alpha \in$ [0, 1] is a hyperparameter
    end while
    Build final unigram LM $\alpha$ to $S$
    return $V$, $\theta$
end procedure
```

Though both BPE and unigram language modeling are subword tokenization algorithms that produce a fixed-size vocabulary for text segmentation, there is a key difference in the method. BPE iteratively merges the most frequent adjacent symbol pairs in the corpus to form new tokens, following a deterministic and greedy approach. In contrast, unigram language modeling initializes with a large vocabulary of candidate substrings and prunes tokens based on their contribution to the overall likelihood under a probabilistic unigram language model, removing those that decrease the model's likelihood until the target vocabulary size is reached.

2.3 Related work

The choice of subword units for acoustic modeling can depend on settings such as high-variability spontaneous speech, noisy environment, low-resource scenario, or cross-lingual speech recognition (Livescu et al., 2012). Different subword modeling techniques have been explored in numerous studies including improved word boundary marker in weighted finite state transducer (WFST)-based decoder for Finnish and Estonian (Smit et al., 2017), pronunciation-assisted subword modeling (PASM) (Xu et al., 2019), acoustic data-driven subword modeling (ADSM) (Zhou et al., 2021), among others. While both PASM and ADSM are reported to outperform BPE-based modeling for ASR, these two approaches are evaluated with only a morphologically poor language English. Thus, it is uncertain how different subword modeling approaches will work for morphologically rich languages.

More recently, phone-based BPE has been introduced for multilingual speech recognition (Yusuyin et al., 2023). However, in a monolingual setting, PBPE obtains similar performance compared to BPE while both BPE and PBPE outperform phone and character-based modeling.

In recent years, Bengali ASR research has primarily focused on addressing the scarcity of datasets through resource development initiatives (Kjartansson et al., 2018; Ahmed et al., 2020; Kibria et al., 2022; Rakib et al., 2023; Samin et al., 2024). Sadeq et al. (2020) addressed the challenge of manual annotation in training ASR systems by proposing a semi-supervised approach for Bangla ASR, leveraging large unpaired audio and text data encoded in an intermediate domain with a novel loss function. Samin et al. (2021) evaluated the LB-ASRTD corpus (Kjartansson et al., 2018), a large-scale publicly available dataset comprising 229 hours, utilizing deep learning-based methods and performing a character-wise error analysis. While earlier studies on Bengali ASR involved phone-based segmentation (Al Amin et al., 2019), the shift towards end-to-end ASR systems has made character-based models more prevalent (Samin et al., 2021). Notably, to the best of our knowledge, there has been no study comparing different subword modeling techniques in the Bengali speech processing domain.

3 Method

We train a convolutional neural network (CNN) based acoustic model for performing the experiments. We extract 21 mel-frequency cepstral coefficients (MFCCs) from each frame of the input signal and feed it to the CNN. The frame length and stride are 30 ms and 15 ms, respectively. We implement the same CNN architecture following the work of Samin et al. (2021), except for introducing a batch normalization

layer in each convolution block to improve optimization stability and reduce internal covariate shift. Moreover, we increase the number of convolutional blocks from 15 to 20, enabling the network to learn deeper hierarchical acoustic representations. The objective of the acoustic model is to predict the subword units based on the CTC loss cri-

Table 1: Six acoustic models are trained with three types of subword units such as character, unigram, and BPE. For BPE segmentation, 500, 1K, 2K, and 3K target tokens are fixed in separate experiments. The models are trained with a CNN architecture on the Bengali SUBAK.KO train set and evaluated on the eval sets of SUBAK.KO, LB-ASRTD, and SHRUTI. TERs (%) and WERs (%) are reported. Bold numbers indicate the best WERs in the corresponding eval sets.

Token type	**# tokens**	**SUBAK.KO eval**		**LB-ASRTD eval**		**SHRUTI eval**	
		TER	**WER**	**TER**	**WER**	**TER**	**WER**
Character	73	5.41	18.89	27.14	66.44	14.31	46.34
Unigram	1000	6.03	16.86	30.32	66.07	15.97	44.40
BPE	500	5.71	17.11	28.15	64.28	14.61	**42.80**
BPE	1000	5.97	16.65	29.32	**63.80**	15.97	43.75
BPE	2000	6.19	16.17	31.99	66.38	17.36	44.58
BPE	3000	6.34	**15.63**	34.11	66.46	18.84	45.77

terion given the audio signal (Graves et al., 2006). We choose either character, BPE, or unigram tokens in individual experiments as our subwords.

We do not perform beam search decoding with a language model since it can have an impact on the final result. The goal of this study is to investigate different subword-based acoustic modeling for a morphologically rich language Bengali, so we exclude the language modeling part. Therefore, greedy search decoding is used to generate the output tokens. In this approach, at each decoding step, the token with the highest predicted probability is selected without considering alternative sequences. This method simplifies decoding and allows us to evaluate the acoustic model's performance independently of any language model influence.

4 Experiment setup

We implement CNN-based acoustic models using the Flashlight toolkit (Kahn et al., 2022). We train our CNNs using SUBAK.KO, an annotated Bangla speech dataset (Kibria et al., 2022). SUBAK.KO is mostly a read speech corpus with 229 hours of read speech and only 12 hours of broadcast speech. We use the same 200-hour long training set, 20-hour long development (dev) set, and 20-hour long evaluation (eval) set following Kibria et al. (2022). Using standard train, dev, and eval sets enables us to compare our strategy to those of the past. For a comprehensive evaluation, we use a 20-hour subset of the large Bangla automatic speech recognition training data (LB-ASRTD) and the 20-hour long full SHRUTI corpus (Kjartansson et al., 2018; Das et al., 2011). Our SUBAK.KO-based ASR model encounters OOV words from LB-ASRTD and SHRUTI out-of-distribution data. Therefore, cross-evaluation assures a more reliable evaluation of various subword modeling algorithms.

We train baseline ASR systems using character and unigram tokens, subsequently contrasting their performance with BPE-based ASR systems. For character-based modeling, we simply use Python programming language to segment a word into individual characters. To build the BPE and unigram-based lexicons, we use the Sentence-piece library (Kudo and Richardson, 2018). For unigram language modeling, we use a fixed token size of 1000. As for BPE, we develop four acoustic models with 500, 1K, 2K, and 3K tokens and compare the results. We use the SUBAK.KO train set as a text corpus to generate the BPE and unigram tokens.

For evaluating ASR models, we employ standard metrics such as the word error rate (WER) and the token error rate (TER). Here, token represents either characters, BPE or Unigram units. Eight graphics processing units (GPUs) with only 12 gigabytes of virtual random access memory each are used to train the models.

5 Results & Discussion

Table 1 presents the WERs and TERs for various subword modeling types and token sizes. On all three evaluation sets, BPE-based acoustic modeling outperforms both character-based and unigram-based modeling in terms of WERs. Unigram modeling achieves lower WERs than character-based segmentation. With the same number of tokens (1000 tokens), unigram modeling cannot surpass the BPE-based approach when dealing with both in-distribution (SUBAK.KO eval) and out-of-distribution (LB-ASRTD and SHRUTI eval) data.

The number of generated tokens is proportional to the number of BPE merge operations. As we increase the number of BPE tokens, the WERs continue to decrease on the SUBAK.KO eval set. Nevertheless, acoustic models trained with 1000 BPE tokens and 500 BPE tokens achieve reduced WERs on the LB-ASRTD and SHRUTI eval sets, respectively. Notably, BPE tokens are generated utilizing the SUBAK.KO train corpus. This implies that as the number of BPE tokens increases, the model becomes overfit on its eval set while performing poorly on the out-of-distribution data. Thus, a target BPE token size of 500 or 1000 is found to be suitable to achieve better generalizability for ASR. This finding conforms to the work of (Gutierrez-Vasques et al., 2023), indicating that morphologically rich languages necessitate fewer BPE merges, leading to a reduced count of BPE tokens. Also, a higher number of BPE merge operations tends to generate longer-length tokens, which resemble words and present similar bottlenecks of word-based tokenization.

With regard to TERs, character-based acoustic modeling exhibits lower error rates than the other two approaches. Furthermore, with the BPE tokens, the TERs tend to continually increase when we increment the number of BPE tokens. Each character represents a token of length one, while BPE tokens undergo multiple merge operations, resulting in longer token lengths. We argue that predicting a token with a shorter length is a comparatively easier task for the acoustic model in contrast to mapping input signal frames to a token with a longer length. This discrepancy can contribute to higher TERs for unigram and BPE-based modeling. However, if a model can accurately predict the long-length tokens of a word, it increases the probability of achieving a lower WER because the number of unigram/BPE tokens in a word is typically fewer than the number of characters in that word.

Figure 1 demonstrates the performance of BPE-based subword modeling with various train corpus sizes. We can observe the positive impact of increasing the amount of acoustic model training data on all three evaluation sets for the BPE-based approach. However, it is worth noting that BPE-based speech recognition systems can still achieve satisfactory performance even in low-resource scenarios.

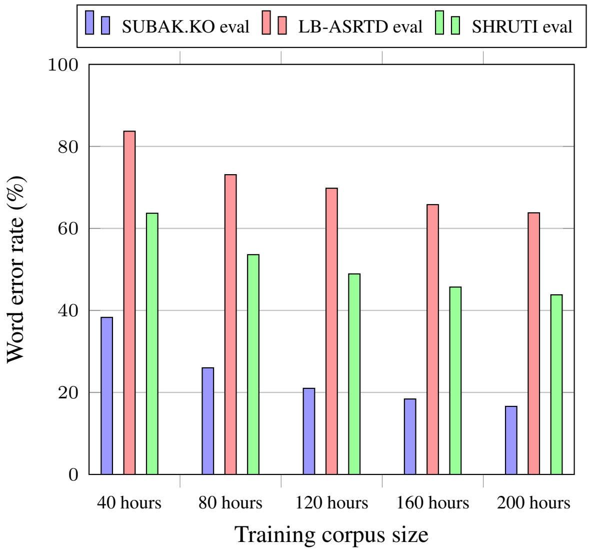

Figure 1: Acoustic models are trained with 1000 BPE tokens on five different SUBAK.KO train subsets (e.g. 40 hours, 80 hours, 120 hours, 160 hours, and 200 hours). SUBAK.KO, LB-ASRTD, and SHRUTI eval sets are used to report the WERs (%)

6 Conclusion & Future Work

In this work, we determine the number of BPE merges in the context of ASR for a morphologically rich language - Bengali and provide intriguing insights into the relationship between BPE merge operations and ASR performance in the presence of OOV words. Furthermore, we provide a comparative analysis for three subword modeling approaches including characters, unigrams, and BPE for ASR. Our empirical study suggests that BPE is a better choice for subword modeling than characters and unigram tokens. Additionally, through cross-dataset evaluation, we find that targeting a token size of approximately 500 or 1000 yields improved generalization and robustness, while excessively high numbers of BPE tokens can result in overfitting. This outcome corresponds with prior linguistic research, suggesting that morphologically rich languages demand fewer BPE merges to yield highly productive BPE tokens, as discussed in (Gutierrez-Vasques et al., 2023).

There are several potential directions for future research. Firstly, instead of generating BPE tokens from the text files of SUBAK.KO train set, a large-scale text corpus could be constructed specifically for this purpose, enabling the generation of BPE tokens from a more extensive and diverse dataset. We hypothesize that this approach could yield superior BPE representations, resulting in enhanced robustness across out-of-domain data, particularly for challenging morphologically rich languages. Secondly, we aim to explore additional morphologically rich languages from diverse language families, as well as languages like English that lack complex morphology, to further evaluate the effectiveness of subword modeling approaches. Lastly, although our study benchmarks convolutional neural networks (CNNs), it would be valuable to investigate state-of-the-art transfer learning algorithms, such as wav2vec 2.0 (Baevski et al., 2020) and HuBERT (Hsu et al., 2021), to determine if BPE subword modeling remains effective with these architectures.

7 Limitations

Although this work investigates BPE subword tokenization method for Bengali ASR for the first time, there are several limitations. First, this study evaluates tokenization only in a monolingual Bengali setting as a representative morphologically rich language. Future work should examine additional such languages to assess generalizability. Second, we investigate the effectiveness of BPE exclusively with CNN-based acoustic models. Exploring alternative architectures remains an open direction. Lastly, this work focuses on comparing BPE with closely related subword approaches (unigram LM and character-based modeling) to isolate and analyze BPE's behavior in morphologically rich Bengali. Therefore, alternative techniques such as PASM (Xu et al., 2019), ADSM (Zhou et al., 2021), and phone-based BPE (Yusuyin et al., 2023), which introduce additional phonetic or acoustic modeling assumptions beyond our scope, are not evaluated in this work.

References

Shafayat Ahmed, Nafis Sadeq, Sudipta Saha Shubha, Md Nahidul Islam, Muhammad Abdullah Adnan, and Mohammad Zuberul Islam. 2020. Preparation of bangla speech corpus from publicly available audio & text. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 6586–6592.

Md Alif Al Amin, Md Towhidul Islam, Shafkat Kibria, and Mohammad Shahidur Rahman. 2019. Continuous bengali speech recognition based on deep neural network. In *2019 international conference on electrical, computer and communication engineering (ECCE)*, pages 1–6. IEEE.

Md Nawab Yousuf Ali, SM Abdullah Al-Mamun, Jugal Krishna Das, and Abu Mohammad Nurannabi. 2008. Morphological analysis of bangla words for universal networking language. In *2008 Third International Conference on Digital Information Management*, pages 532–537. IEEE.

Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in neural information processing systems*, 33:12449–12460.

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.

William Chan, Navdeep Jaitly, Quoc Le, and Oriol Vinyals. 2016. Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4960–4964.

Biswajit Das, Sandipan Mandal, and Pabitra Mitra. 2011. Bengali speech corpus for continuous auutomatic speech recognition system. In *2011 International conference on speech database and assessments (Oriental COCOSDA)*, pages 51–55. IEEE.

Philip Gage. 1994. A new algorithm for data compression. *The C Users Journal*, 12(2):23–38.

Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. 2006. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd international conference on Machine learning*, pages 369–376.

Ximena Gutierrez-Vasques, Christian Bentz, and Tanja Samardžić. 2023. Languages through the looking glass of bpe compression. *Computational Linguistics*, pages 1–59.

Ximena Gutierrez-Vasques, Christian Bentz, Olga Sozinova, and Tanja Samardzic. 2021. From characters to words: the turning point of bpe merges. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3454–3468.

David F Harwath and James R Glass. 2014. Speech recognition without a lexicon-bridging the gap between graphemic and phonetic systems. In *INTERSPEECH*, pages 2655–2659.

Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. 2021. Hubert: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:3451–3460.

Jacob D Kahn, Vineel Pratap, Tatiana Likhomanenko, Qiantong Xu, Awni Hannun, Jeff Cai, Paden Tomasello, Ann Lee, Edouard Grave, Gilad Avidov, and 1 others. 2022. Flashlight: Enabling innovation in tools for machine learning. In *International Conference on Machine Learning*, pages 10557–10574. PMLR.

Shafkat Kibria, Ahnaf Mozib Samin, M Humayon Kobir, M Shahidur Rahman, M Reza Selim, and M Zafar Iqbal. 2022. Bangladeshi bangla speech corpus for automatic speech recognition research. *Speech Communication*, 136:84–97.

Oddur Kjartansson, Supheakmungkol Sarin, Knot Pipatsrisawat, Martin Jansche, and Linne Ha. 2018. Crowd-sourced speech corpora for javanese, sundanese, sinhala, nepali, and bangladeshi bengali. In *6th Workshop on Spoken Language Technologies for Under-Resourced Languages (SLTU 2018)*. ISCA.

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.

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.

Karen Livescu, Eric Fosler-Lussier, and Florian Metze. 2012. Subword modeling for automatic speech recognition: Past, present, and emerging approaches. *IEEE Signal Processing Magazine*, 29(6):44–57.

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

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training. Technical report, OpenAI. OpenAI Technical Report.

Fazle Rabbi Rakib, Souhardya Saha Dip, Samiul Alam, Nazia Tasnim, Md Istiak Hossain Shihab, Md Nazmuddoha Ansary, Syed Mobassir Hossen, Marsia Haque Meghla, Mamunur Mamun, Farig Sadeque, and 1 others. 2023. Ood-speech: A large bengali speech recognition dataset for out-of-distribution benchmarking. *Proc. Interspeech 2023*.

Nafis Sadeq, Nafis Tahmid Chowdhury, Farhan Tanvir Utshaw, Shafayat Ahmed, and Muhammad Abdullah Adnan. 2020. Improving end-to-end Bangla speech recognition with semi-supervised training. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1875–1883, Online. Association for Computational Linguistics.

Ahnaf Mozib Samin, M Humayon Kobir, Shafkat Kibria, and M Shahidur Rahman. 2021. Deep learning based large vocabulary continuous speech recognition of an under-resourced language bangladeshi bangla. *Acoustical Science and Technology*, 42(5):252–260.

Ahnaf Mozib Samin, M Humayon Kobir, Md Mushtaq Shahriyar Rafee, M Firoz Ahmed, Mehedi Hasan, Partha Ghosh, Shafkat Kibria, and M Shahidur Rahman. 2024. Banspeech: A multi-domain bangla speech recognition benchmark toward robust performance in challenging conditions. *IEEE Access*, 12:34527–34538.

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

Peter Smit, Sami Virpioja, and Mikko Kurimo. 2017. Improved subword modeling for wfst-based speech recognition. In *INTERSPEECH*, pages 2551–2555. International Speech Communication Association.

Gabriel Synnaeve, Qiantong Xu, Jacob Kahn, Tatiana Likhomanenko, Edouard Grave, Vineel Pratap, Anuroop Sriram, Vitaliy Liptchinsky, and Ronan Collobert. 2020. End-to-end asr: from supervised to semi-supervised learning with modern architectures. In *ICML 2020 Workshop on Self-supervision in Audio and Speech*.

Hainan Xu, Shuoyang Ding, and Shinji Watanabe. 2019. Improving end-to-end speech recognition with pronunciation-assisted sub-word modeling. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7110–7114. IEEE.

Saierdaer Yusuyin, Hao Huang, Junhua Liu, and Cong Liu. 2023. Investigation into phone-based subword units for multilingual end-to-end speech recognition. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.

Wei Zhou, Mohammad Zeineldeen, Zuoyun Zheng, Ralf Schlüter, and Hermann Ney. 2021. Acoustic data-driven subword modeling for end-to-end speech recognition. *arXiv preprint arXiv:2104.09106*.