# BPE beyond Word Boundary: How NOT to use Multi Word Expressions in Neural Machine Translation

**Dipesh Kumar***
Indian Institute of Technology (BHU) / Varanasi
dipesh.kumar.cse17@iitbhu.ac.in

**Avijit Thawani***
University of Southern California / Los Angeles
Information Sciences Institute / Marina del Rey
thawani@usc.edu

## Abstract

BPE tokenization merges characters into longer tokens by finding frequently occurring **contiguous** patterns **within** the word boundary. An intuitive relaxation would be to extend a BPE vocabulary with multi-word expressions (MWEs): bigrams ($in\_a$), trigrams ($out\_of\_the$), and skip-grams ($he{\cdot}his$). In the context of Neural Machine Translation (NMT), we replace the least frequent subword/whole-word tokens with the most frequent MWEs. We find that these modifications to BPE end up hurting the model, resulting in a net drop of BLEU and chrF scores across two language pairs. We observe that naively extending BPE beyond word boundaries results in incoherent tokens which are themselves better represented as individual words. Moreover, we find that Pointwise Mutual Information (PMI) instead of frequency finds better MWEs (e.g., $New\_York$, $Statue\_of\_Liberty$, $neither{\cdot}nor$) which consistently improves translation performance. We release all code at https://github.com/pegasus-lynx/mwe-bpe.

## 1 Introduction

Subword tokenization algorithms like Byte Pair Encoding (BPE) (Sennrich et al., 2016) group together frequently occurring patterns, such as *-ing* or *-ly*, into individual tokens. The success of subword tokenization points to the benefit in modeling longer patterns, even though any given text can be represented simply as a sequence of characters. This paper stretches the motivation further by allowing BPE to cross word boundaries. In the context of NMT, we find that the straightforward way to find MWEs by BPE (sorted by frequency) hurts performance whereas sorting by PMI scores improves scores. We hypothesize and discuss a reason for these observations and provide further recommendations on using MWEs with BPE.

N-gram tokens have been used in traditional NLP for a long time and with much success. For example (Table 1), the bigram *New York* can be a concise yet useful feature in a Named Entity Recognition task. Similarly, a Spanish-English Machine Translation (MT) model might benefit from having the bigram *te amo* or its trigram translation *I love you* in its vocabulary. Finally, a model's vocabulary could even extend to non-contiguous tokens or k-skip-n-grams such as *neither · nor*. This token reappears in several contexts e.g. *neither tea nor coffee* and *neither here nor there* (underlined words replace the · skip).

| | |
|---|---|
| **Raw** | He lives in New York . |
| **Tok** | He_ lives_ in_ New_York_ ._ |
| **Raw** | I love the Statue of Liberty! |
| **Tok** | I_ love_ the_ Statue_of_Liberty_ !_ |
| **Raw** | She lost her bag . |
| **Tok** | She_ · her_ lost_ <SKIP> bag_ ._ |

Table 1: Example tokenizations of MWEs (bigrams, trigrams, skip-grams) in our implementation. Raw = original sentence, Tok = tokenized form. Typical BPE tokens are colored yellow and MWEs are colored green.

This paper experiments with two ways to expand BPE with MWEs for the task of NMT. Concretely, we promise the following contributions:

1. We find, counter-intuitively, that the straightforward frequency-based BPE, when applied beyond words, performs worse than baseline on NMT across two language pairs (§3).

2. We hypothesize that this negative result is caused by the constituents of such high frequency MWEs (e.g. $in\_the$) combining in many diverse ways, rendering such tokens incoherent (§4.1).

3. We show that PMI-based BPE for MWEs reverses the drop and improves BLEU scores. We offer more recommendations on where and how to use MWEs with BPE (§4.2).

* Equal Contribution.

*Proceedings of the Third Workshop on Insights from Negative Results in NLP*, pages 172 - 179
May 26, 2022 ©2022 Association for Computational Linguistics

| **Lang. Pair** | Hi → En | | | | De → En | | | |
|---|---|---|---|---|---|---|---|---|
| **Split** | Dev | | Test | | Dev | | Test | |
| | sacre | chrF | sacre | chrF | sacre | chrF | sacre | chrF |
| **Metric** | BLEU | $\beta = 2$ | BLEU | $\beta = 2$ | BLEU | $\beta = 2$ | BLEU | $\beta = 2$ |
| **Baseline** | 20.8 | 49.5 | 22.0 | 52.3 | 39.1 | **62.4** | 35.6 | 59.1 |
| **Unigram** | 19.5 | 49.0 | 21.2 | 51.5 | 36.5 | 60.3 | 32.4 | 56.8 |
| **BPE+ngms** | 19.5 | 49.0 | 21.2 | 51.6 | 38.7 | 62.2 | 35.3 | 58.9 |
| **BPE+n/sgms** | 18.4 | 48.1 | 20.7 | 51.3 | 38.4 | 62.1 | 35.2 | 58.9 |
| PMI methods | | | | | | | | |
| **Bigrams** | 20.6 | 49.2 | **22.2** | **52.6** | **39.1** | **62.4** | 35.8 | **59.3** |
| **Trigrams** | 20.7 | 49.5 | 22.0 | 52.3 | 39.0 | 62.2 | 35.7 | 59.0 |
| **N-grams** | **21.2** | **50.0** | 22.1 | **52.6** | 38.9 | 62.3 | 35.8 | 59.1 |
| **Skip-grams** | 20.6 | 49.9 | 22.1 | 52.4 | 38.7 | 62.1 | **35.9** | 59.2 |

Table 2: Different methods of adding MWEs to a BPE vocabulary on NMT across two language pairs.

## 2 Methods

MWEs have been commonly used in traditional NLP but rarely in the age of transformers and subword vocabularies. Here we describe two kinds of ways to add MWEs to a BPE vocabulary.

### 2.1 BPE beyond words

Our baseline is the vanilla BPE tokenization scheme which starts from characters and iteratively adds the most frequent subwords to vocabulary. An intuitive extension to BPE is **BPE+ngms**, i.e., allowing BPE to choose between not just adding subwords but also frequently occurring n-grams (e.g., of_the appears at $163^{rd}$ position in vocabulary). This paper limits n-grams to bigrams and trigrams.

Besides continuous multi-word expressions, we also experiment with discontinuous MWEs, i.e., k-skip-n-grams, which we refer to concisely as skip-grams. In particular, we focus on 1-skip-3-grams, e.g., *neither · nor*, *I · you*. We replace a 1-skip-3-gram ($w_1 \cdot w_2$) occurrence with ($w_{12}$ · `<SKIP>`) where $w_{12}$ is a new token representing the occurrence of this specific 1-skip-3-gram, and `<SKIP>` is another new token but shared by all skip-grams to indicate that the skip-gram ends here. The last row of Table 1 shows an example tokenization with skip-grams. In **BPE+n/sgms**, we allow frequent skip-grams (e.g., ( · ); neither · nor ) to also be part of the vocabulary.

### 2.2 Adding MWEs with PMI

As hinted in Section 1, the intuitive extension to BPE does not work well in practice. Instead of raw frequency, here we find MWEs using a common technique of finding word collocations: Pointwise Mutual Information (PMI), which is a measure of the association between two word types in text. We calculate PMI of n-grams as:

$$PMI(a_1, ..., a_n) = \log\left(\frac{P(a_1, ..., a_n)}{\prod_{i=1}^{n} P(a_i)}\right)$$

where $a_i$ are unigrams (words) from the corpus; $P(a_i)$ denote their independent probabilities; and $P(a_i, ...a_n)$ denotes joint probability of n-grams. In this paper, we report experiments with only **Bigrams** ($n = 2$), **Trigrams** ($n = 3$), and their combination **N-grams**.

We also experiment with **Skip-grams** or 1-skip-3-grams ($w_1 \cdot w_2$) from our corpus in the same way as bigrams ($w_1 w_2$), ordered by PMI. We identify candidate word pairs separated by one word (which we depict by · ) and sort them based on PMI scores, some of which are deemed good enough to replace the least frequent subwords in the BPE vocabulary.

We find that the skip-grams obtained by simply ordering by PMI are often better suited to be trigrams, e.g., the · in *Statue · Liberty*, a high-ranked candidate skip-gram, is almost always *of*. To disentangle such skip-grams, we filter out candidates where the middle (skipped) word has a spread-out distribution: the skipped word in *I · you* could be replaced with several words like *love*, *hate*, or *miss*. In practice, we filter these by enforcing (1) a lower limit (15) on the number of unique words which replace the · token, and (2) an upper limit on the probability (10%) of the most frequently occurring skipped token for the particular skip-gram.

## 3 Datasets

We use the IIT Bombay Hindi-English parallel corpus v3.0 (Kunchukuttan et al., 2018), tokenized using IndicNLPLibrary (Kunchukuttan, 2020) and Moses Tokenizer (Koehn et al., 2007) respectively. The Train : Dev : Test splits have $1.6M : 0.5K : 2.3K$ sentences respectively.

For German-English, the datasets are retreived from the News Translation task of WMT2019 (Barrault et al., 2019). The Train : Dev : Test splits have $4.5M : 3K : 2K$ sentences respectively.

While we use the originally mentioned training set for our main results in Table 2, we found several noisy sentence pairs in the training dataset (the dev and test set were clean). Some such sentences had English characters (latin alphabet) in the source (Hindi) side and others had non-English characters on the target (English) side. We filtered out 250K sentence pairs where either the source side had non-Hindi characters or the target side had non-English characters, wherein we count the following near-universal symbols as part of either language: $.,()[]!:-"';<>?\&\breve{}@$

## 4 Experiments

While MWEs can augment the subword vocabulary of any NLP model, this short paper focuses on the task of NMT. Following Gowda and May (2020), we fix the transformer architecture (Vaswani et al., 2017) and train models with different vocabularies from scratch.

Our baseline vocabulary is BPE with 8K subword tokens for Hi-En and 16K for De-En. Each of our methods maintains the same vocabulary size, replacing the least frequently occurring subwords with corresponding n-grams or skip-grams. We show representative MWEs learned from corpora in Table 4 alongside the coverage of (PMI) MWEs across language pairs.

We also compare with a Unigram (Kudo, 2018) SentencePiece vocabulary of 8K tokens each on source and target sides, with *split_by_whitespace* flag set to `false` (Kudo and Richardson, 2018). This allows the Unigram method to go beyond the word boundary and add n-grams to its vocabulary.

Our NMT model is a 6 layer transformer encoder-decoder (Vaswani et al., 2017) that has 8 attention heads, 512 hidden vector units, and a feed forward intermediate size of 2048, with GELU activation. We use label smoothing at 0.1, and a dropout rate of 0.1. We use the Adam optimizer with a controlled learning rate that warms up for 16K steps followed by a decay rate recommended for training transformer models. We trim longer sequences to a maximum of 512 tokens after BPE. Each model is trained from scratch, and the hyperparameters (per language pair) are chosen by grid search to optimize the baseline validation BLEU.

We train all models for up to $100K$ steps (batch size $= 24K$ tokens) and report sacreBLEU (Post, 2018) and chrF ($\beta = 2$) scores (Popović, 2015).

The number of tokens replaced in the original BPE vocabulary with a corresponding MWE ordered by PMI, is also a hyperparameter optimized by grid search between 1.25% to 10% of the vocabulary size (Hi-En models performing best when 1.25% tokens were replaced and De-En models performing best at 2.5% for Bigrams/Trigrams and 5% for Skipgrams). We make sure to not replace any rare base characters like $Q$ or @.

For ablations (Section 5.2) with limited compute budget, we train Hi-En models for up to 200K steps.We apply a patience of 10 validations, each 1000 update steps apart. To decode, we average the best 3 checkpoints, and use a beam size of 4 with length penalty of 0.6. We use NLCodec and RTG libraries (Gowda et al., 2021) and contribute our extensions to them as well.

## 5 Results and Discussion

Table 2 shows our main results. We find that naively extending BPE beyond words harms the model, and Unigram likewise fails to consistently outperform the baseline. On the other hand, adding MWEs using PMI gives the best performance across language pairs and metrics.

Moreover, since the methods of extracting MWEs is purely emprirical and is language agnostic, the results and observations can be extended for different language pairs.

We now attempt to reason why BPE fails beyond word boundaries in its vanilla form, and why switching to PMI solves the problem. We also study where does it help the most to add MWEs. Unless noted otherwise, the analysis is reported on the Hi-En dataset.

### 5.1 Words combine in Diverse ways

Empirically, we observe (Table 2) that BPE with high frequency MWE tokens sees a drop in performance whereas the PMI counterpart as well as the original baseline (within word boundary) performs

| | Train | Dev | Test |
|---|---|---|---|
| **Hi-En** | IITB-Training (1.3M) | IITB-Dev (0.5K) | IITB-Test (2.5K) |
| **De-En** | Europarl v10 (1.8M)<br>WMT13CommonCrawl (2.4M)<br>NewsCommentary v14 (0.3M) | NewsTest18 (3K) | NewsTest19 (2K) |

Table 3: Training, validation and testing datasets along with sentence count in each set.

| | from Hi-En to De-En | from De-En to Hi-En |
|---|---|---|
| **Bi** | 1.55% | 1.30% |
| **Tri** | 0.30% | 0.40% |
| **Skip** | **13.34%** | **13.45%** |

| Bigrams | Trigrams | Skip-Grams | Freq |
|---|---|---|---|
| per cent | New York City | the · of | of the |
| New York | European Central Bank | a · of | do not |
| Prime Minister | Italian Prime Minister | ( · ) | they are |
| Middle East | behind closed doors | was · to | as well as |
| United Nations | former Prime Minister | not · to | one of the |

Table 4: **Left**: Coverage of the top 5 most frequent English MWEs (PMI-based), extracted from the first language pair and (coverage) evaluated over the second. Coverage of a token is defined as the fraction of target (English) sentences containing the token. **Right**: The top five MWEs of each type (PMI except when labelled Freq).

well. What then happens at the word boundary that the BPE algorithm stops working? We hypothesize that this is the result of words combining in more diverse ways than subwords.

BPE beyond word boundary adds frequently occuring n-grams to its vocabulary such as *in_the* which occurs in over a tenth of all test sentences. Despite adding it as a separate token to the vocabulary, the average BLEU on this subset of test sentences drops compared to the baseline (20.0 vs 21.8)! One factor for this result could be that the constituents of *in_the* combine in more ways than one. The word *in* appears as the ending of over 30 n-grams (*that_in*, *was_in*, …) and the word *the* appears as the beginning of 200 other n-grams (*the_people*, *the_first*, …) - all of which combine to a total of over another tenth of the test set, more than the frequency of *in_the* itself.

Such versatile combinatorics is rarely observed at the subword level. Suffixes like *ing* almost never appear as prefixes whereas prefixes like *de* almost never appear as suffixes. When such subwords combine to form longer tokens, they generally retain a coherent meaning, unlike n-grams like *in_the*. Finally, this hypothesis may explain why MWEs ordered by PMI help improve MT scores – they are by definition units that co-occur as a coherent unit. Indeed, the MWEs thus found (e.g. *New_York*, *per_cent*) include constituents which exclusively form only these tokens.

To summarize, we argue that BPE stops working at word boundaries because word pairs rarely, unlike subwords, combine into meaningful units that deserve a unique representation. We find convincing arguments from sentence-level BLEU scores and the number of different ways the constituents of different tokens occur, more of which are reported in supplementary materials.

## 5.2 Where do MWEs help NMT?

Here, we conduct ablations for the PMI method (on a smaller batch size of 1K tokens, on the Hi-En dataset) to determine whether MWEs help more for machine translation on the source side (Hi), on the target side (En), or both? Table 2 reports on the 'both' setting but here we revisit this design choice. Table 5 reports BLEU scores with each such variant. Bold-faced cells indicate the best performing (on dev set) variant for every row. We observe that continuous MWEs (bigrams and trigrams) benefit more on the source-side whereas discontinuous MWEs (skip-grams) help the most when applied to both source and target side. Note that, since De-En has been usually used in a triple shared vocabulary setting, we followed the same and thereby it must always follow the 'both' model.

Finally, we show in Figure 1 some representative examples of sentences with MWEs (particularly, the skip-grams) from the PMI-BPE Hi-En model's vocabulary. The first two rows show examples where the skip-gram indeed occurred in the reference, hence it helped the model. The last row shows how the model overuses the skip-gram, i.e. using skip-gram instead of separate tokens, and gets a

| **Source** | **Reference** | **Baseline** | **Skip-Gram** | **Helps/ Hurts?** |
|---|---|---|---|---|
| यह परियोजना पूरे यूरोपीय महाद्वीप की ऊर्जा सुरक्षा का एक मुख्य तत्व है। | **This** project **is** a key element of energy security of the whole European continent. | The project is a major element of the energy security of the entire European continent. | **This** project **is** the main element of energy security of the entire European continent. | Helps |
| यह पुरस्कार व्यापक रूप से ... | **This** award **is** widely considered the ... | The award is widely recognized as the ... | **This** award **is** widely regarded as the ... | Helps |
| यह क्षेत्र एक ... से भरा हुआ है, ... | The area is filled with ... | The region is filled with a ... | **This** area **is** full of ... | Hurts |

Figure 1: Qualitative error analysis over Hi-En test set, showing examples comparing the Baseline and the Skip-Gram augmented model, where the skip-gram (**This · is**) occurs in the latter's predictions.

| | **Target (En)** | **Source (Hi)** | **Both** |
|---|---|---|---|
| **Bi** | 14.4 / 14.8 | **15.9 / 16.0** | 15.8 / 15.3 |
| **Tri** | 14.7 / 15.4 | **15.5 / 15.5** | 15.4 / 15.2 |
| **Skip** | 15.3 / 15.2 | 15.1 / 15.1 | **15.5 / 15.0** |

Table 5: Do MWEs help more when added to the source-side, the target-side or both? Each cell reports Dev/Test BLEU scores over Hi-En dataset only. Baseline scores without MWEs are 15.6 / 14.4 respectively.

translation wrong thus hurting the score as the reference sentence does not use the skip-gram. We note that BLEU itself relies only on the presence or absence of contiguous n-grams, and may unfairly penalize paraphrased outputs such as these.

## 6 Related Work

Attempts at merging NMT with MWEs typically include pairing up the network with a phrase based SMT system (Wang et al., 2017; Park and Tsvetkov, 2019; Lample et al., 2018) and hierarchical phrases are expressive enough to cover discontinuous MWEs (Chiang, 2007). Zaninello and Birch (2020) add manually annotated MWEs aligned across the source and target language (En-It). However, this might not work for low resource languages, hence we extract MWEs automatically with PMI. They count discontinuous MWEs, one of our main contributions, among future work.

Multi-word tokens have a proven track record in NLP. Skip-gram tokens, for instance, have already been used in phrase-based machine translation (Lample et al., 2018; Park and Tsvetkov, 2019; Wang et al., 2017) to tackle cases where certain phrases in a source language (*duonianlai* in Chinese) are better represented as skip-grams in a target language (*over the last · years* in English) (Chiang, 2007). Our work revisits these ideas and adapts them to a transformer-based NLP model relying on subword segmentation. There also exists prior work on defining, counting, and evaluating k-skip-n-grams (Guthrie et al., 2006; Pickhardt et al., 2014; Ptaszynski et al., 2014), although unrelated to the task of NMT. Finally, readers interested in other applications of extracting MWEs via PMI scores may refer to Levine et al. (2021) where similar techniques are used to efficiently mask tokens while pretraining BERT (Devlin et al., 2019).

## 7 Conclusion

This paper systematically studies the impact of extending a BPE vocabulary with multi-word expressions for neural machine translation. Our results point to the vast unexplored scope of different granularities of tokenization that can be exploited by NLP systems. Notably, our methods extend to not only longer contiguous tokens like n-grams but also skip-grams, which have been relatively unexplored with transformer-based NLP.

In future work, we intend to compare our PMI-based methods to human-annotated MWEs as well as to recent workarounds to interfering tokenization schemes such as subword regularization or BPE dropout (Provilkov et al., 2020). We also wish to extend experiments to NLP tasks beyond NMT, and the scope of our tokens to, say, variable-skip-grams which allow for any number of skips.

## References

Loïc Barrault, Ondřej Bojar, Marta R. Costa-jussà, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Philipp Koehn, Shervin Malmasi, Christof Monz, Mathias Müller, Santanu Pal, Matt Post, and Marcos Zampieri. 2019. Findings of the 2019 conference on machine translation (WMT19). In *Proceedings of the Fourth Con-*

ference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 1–61, Florence, Italy. Association for Computational Linguistics.

Jan A. Botha and Phil Blunsom. 2013. Adaptor Grammars for learning non-concatenative morphology. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 345–356, Seattle, Washington, USA. Association for Computational Linguistics.

David Chiang. 2007. Hierarchical phrase-based translation. *Computational Linguistics*, 33(2):201–228.

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.

Thamme Gowda and Jonathan May. 2020. Finding the optimal vocabulary size for neural machine translation. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3955–3964, Online. Association for Computational Linguistics.

Thamme Gowda, Zhao Zhang, Chris A Mattmann, and Jonathan May. 2021. Many-to-english machine translation tools, data, and pretrained models.

David Guthrie, Ben Allison, Wei Liu, Louise Guthrie, and Yorick Wilks. 2006. A closer look at skip-gram modelling. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*, Genoa, Italy. European Language Resources Association (ELRA).

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic. 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, Brussels, Belgium. Association for Computational Linguistics.

Anoop Kunchukuttan. 2020. The IndicNLP Library. `https://github.com/anoopkunchukuttan/indic_nlp_library/blob/master/docs/indicnlp.pdf`.

Anoop Kunchukuttan, Pratik Mehta, and Pushpak Bhattacharyya. 2018. The IIT Bombay English-Hindi parallel corpus. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Guillaume Lample, Myle Ott, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018. Phrase-based & neural unsupervised machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5039–5049, Brussels, Belgium. Association for Computational Linguistics.

Yoav Levine, Barak Lenz, Opher Lieber, Omri Abend, Kevin Leyton-Brown, Moshe Tennenholtz, and Yoav Shoham. 2021. {PMI}-masking: Principled masking of correlated spans. In *International Conference on Learning Representations*.

Chan Young Park and Yulia Tsvetkov. 2019. Learning to generate word- and phrase-embeddings for efficient phrase-based neural machine translation. In *Proceedings of the 3rd Workshop on Neural Generation and Translation*, pages 241–248, Hong Kong. Association for Computational Linguistics.

Rene Pickhardt, Thomas Gottron, Martin Körner, Paul Georg Wagner, Till Speicher, and Steffen Staab. 2014. A generalized language model as the combination of skipped n-grams and modified Kneser Ney smoothing. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1145–1154, Baltimore, Maryland. Association for Computational Linguistics.

Maja Popović. 2015. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395, Lisbon, Portugal. Association for Computational Linguistics.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

Ivan Provilkov, Dmitrii Emelianenko, and Elena Voita. 2020. BPE-dropout: Simple and effective subword regularization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1882–1892, Online. Association for Computational Linguistics.

Michal Ptaszynski, Fumito Masui, Rafal Rzepka, and Kenji Araki. 2014. First glance on pattern-based language modeling. *Language Acquisition and Understanding Research Group Technical Reports*.

Michal Ptaszynski, Rafal Rzepka, and Yoshio Momouchi. 2011. Language combinatorics: A sentence pattern extraction architecture based on combinatorial explosion. *International Journal of Computational Linguistics*, 2(1):24–36.

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

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Xing Wang, Zhaopeng Tu, Deyi Xiong, and Min Zhang. 2017. Translating phrases in neural machine translation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1421–1431, Copenhagen, Denmark. Association for Computational Linguistics.

Andrea Zaninello and Alexandra Birch. 2020. Multiword expression aware neural machine translation. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 3816–3825, Marseille, France. European Language Resources Association.

## A Visualizing the top-scoring MWEs

We already report highest scoring English MWEs throughout the paper, particularly in Table 4. In Figure 2, we enumerate similarly the highest scoring bigrams, trigrams, and skip-grams from the other two languages: German and Hindi.

## B Scope of MWEs

As the name suggests, MWEs include only word level expressions i.e., *each constituent should be a whole word*. This is a less expressive but more intuitive approach to going beyond word boundaries with BPE. For example, our implementation does not allow for tokens that combine the ending of one word and the beginning of another.

Note that our implementation also allows for variable length skip-grams (Ptaszynski et al., 2011), represented as $(w_1 * w_2)$. Instead of skipping a single token, we can allow skipping any number of tokens and still map to the same skip-gram, e.g., *neither* $*$ *nor* → *neither do I drink nor do I smoke*. Such tokens would be much more expressive but also much computationally expensive to find, and would require some simplifying assumptions such as disallowing nested skip-grams. We leave such experiments to future work.

Note that we do not merge bigrams, trigrams, and skip-grams. PMI scores across n-grams and skip-grams are not comparable, hence they can not be combined in a straightforward way. Such an amalgamation may indeed give an even bigger boost but requires grid search over multiple hyperparameters corresponding to the fraction of each kind of MWE to be included. Such experiments warrant an extensive compute budget, so we leave this to future work.

We wish to implement even newer forms of tokenization, particularly extending skip-gram tokens. While this paper limits skip-grams to only act at the word level, one could also imagine character or subword level skip-grams, such as *r-n* serving as a skip-gram common to both *run* and *ran*. Finally, k-skip-n-gram tokens need not be limited to a fixed k, allowing for a variable number of tokens to be skipped, similar to a hierarchical phrase translation system (Chiang, 2007). Such variable length skips can also be useful at the character level, e.g., *k-t-b* as a skip-gram for both *kitaab* and *kutub* (Botha and Blunsom, 2013).

| German | | | Hindi | | |
|---|---|---|---|---|---|
| **Bi/Tri Grams (Freq)** | **Bi/Tri Grams (PMI)** | **Skip-Grams** | **Bi/Tri Grams (Freq)** | **Bi/Tri grams (PMI)** | **Skip-Grams** |
| in der<br>zu den<br>in Bezug auf | Vereinten Nationen<br>ums Leben<br>kurze Zeit später | die · des<br>bis · Prozent<br>Zwischen · und | के लिए<br>नहीं किया<br>जिन लोगों ने | मोबाइल फोन<br>क्रेडिट कार्ड<br>सुभाष चन्द्र बोस | का · किया<br>कोई · नहीं<br>इस · में |

Figure 2: Top scoring multi-word expressions extracted from the training corpora.