BARTpho: Pre-trained Sequence-to-Sequence Models for Vietnamese

Nguyen Luong Tran, Duong Minh Le, Dat Quoc Nguyen

VinAI Research, Hanoi, Vietnam

{v.nguyentl12, v.duonglm1, v.datnq9}@vinai.io

Abstract

We present BARTpho with two versions, BARTpho\textsubscript{syllable} and BARTpho\textsubscript{word}, which are the first public large-scale monolingual sequence-to-sequence models pre-trained for Vietnamese. BARTpho uses the “large” architecture and the pre-training scheme of the sequence-to-sequence denoising autoencoder BART, thus it is especially suitable for generative NLP tasks. We conduct experiments to compare our BARTpho with its competitor mBART on a downstream task of Vietnamese text summarization and show that: in both automatic and human evaluations, BARTpho outperforms the strong baseline mBART and improves the state-of-the-art. We further evaluate and compare BARTpho and mBART on the Vietnamese capitalization and punctuation restoration tasks and also find that BARTpho is more effective than mBART on these two tasks. We publicly release BARTpho to facilitate future research and applications of generative Vietnamese NLP tasks.

Index Terms: BARTpho; Sequence-to-Sequence; Vietnamese; Pre-trained models; Text summarization; Capitalization; Punctuation restoration.

1. Introduction

The masked language model BERT [1] and its variants, pre-trained on large-scale corpora, help improve the state-of-the-art (SOTA) performances of various natural language understanding (NLU) tasks. However, due to a bidirectionality nature, it might be difficult to directly apply those pre-trained language models to natural language generation tasks [2]. Therefore, pre-trained sequence-to-sequence (seq2seq) models are proposed to handle this issue [3, 4, 5, 6, 7, 8]. The success of these pre-trained seq2seq models has largely been limited to the English language. From a societal, cultural, linguistic, cognitive and machine learning perspective [9], it is worth investigating pre-trained seq2seq models for languages other than English. For other languages, one could employ existing pre-trained multilingual seq2seq models [10, 11, 12] or retrain language-specific models using the proposed seq2seq architectures [13, 14]. Note that retraining a language-specific model might be preferable as dedicated language-specific models still outperform multilingual ones [15].

Regarding Vietnamese, to the best of our knowledge, there is not an existing monolingual seq2seq model pre-trained for Vietnamese. In addition, another concern is that all publicly available pre-trained multilingual seq2seq models are not aware of the linguistic characteristic difference between Vietnamese syllables and word tokens. This comes from the fact that when written in Vietnamese, in addition to marking word boundaries, the white space is also used to separate syllables that constitute words.\footnote{Note that 85\% of Vietnamese word types are composed of at least two syllables [16].}

We are researchers forms a 4-word text “Chúng tôi là những nghiên cứu viên”. Without applying a Vietnamese word segmenter, those pre-trained multilingual seq2seq models directly apply Byte-Pair encoding models [17, 18] to the syllable-level Vietnamese pre-training data. Therefore, it is worth investigating the influence of word segmentation on seq2seq pre-training for Vietnamese.

In this paper, we introduce BARTpho with two versions—BARTpho\textsubscript{syllable} and BARTpho\textsubscript{word}—the first large-scale monolingual seq2seq models pre-trained for Vietnamese, which are based on the seq2seq denoising autoencoder BART [4]. The difference between our two BARTpho versions is that they take different types of input texts: a syllable level for BARTpho\textsubscript{syllable} vs. a word level for BARTpho\textsubscript{word}. We compare BARTpho with mBART [10]—a multilingual variant of BART—on a downstream task of Vietnamese text summarization, and we find that our BARTpho models outperform mBART in both automatic and human evaluations, and help produce a new SOTA performance, thus showing the effectiveness of large-scale monolingual seq2seq pre-training for Vietnamese. We also evaluate and compare BARTpho and mBART on the Vietnamese capitalization and punctuation restoration tasks and find that BARTpho helps produce better performance results than mBART. In all three evaluation tasks, we find that BARTpho\textsubscript{word} does better than BARTpho\textsubscript{syllable}, showing the positive influence of Vietnamese word segmentation towards seq2seq pre-training.

We publicly release our BARTpho models at https://github.com/VinAIResearch/BARTpho, which can be used with popular libraries fairseq [19] and transformers [20]. We hope that our BARTpho can serve as a strong baseline for future research and applications of generative natural language processing (NLP) tasks for Vietnamese.

2. Related work

PhoBERT [15] is the first public large-scale monolingual language model pre-trained for Vietnamese, which helps obtain state-of-the-art performances on various downstream Vietnamese NLP/NLU tasks [21, 22, 23, 24, 25]. PhoBERT is pre-trained on a 20GB word-level corpus of Vietnamese texts, using the RoBERTa pre-training approach [26] that optimizes BERT for more robust performance. Following PhoBERT, there are also public monolingual language models for Vietnamese such as viBERT and vELECTRA [27], which are based on BERT and ELECTRA pre-training approaches [1, 28] and pre-trained on syllable-level Vietnamese text corpora. Following Rothe et al. [29] who leverage pre-trained language model checkpoints for sequence generation tasks, Nguyen et al. [30] conduct an empirical study and show that PhoBERT helps produce better performance results than viBERT for a downstream task of Vietnamese abstractive summarization.

Our BARTpho is based on BART. We employ BART because it helps produce the strongest performances on downstream tasks in comparison to other pre-trained seq2seq models.
from transformers import AutoModel, AutoTokenizer

# BARTpho_syllable
tokenizer = AutoTokenizer.from_pretrained("vinai/bartpho-syllable")
bartpho_syllable = AutoModel.from_pretrained("vinai/bartpho-syllable")
input_text = 'Chúng tôi là những nghiên cứu viên'
input_ids = tokenizer(input_text, return_tensors='pt')
features = bartpho_syllable(**input_ids)

# BARTpho_word
tokenizer = AutoTokenizer.from_pretrained("vinai/bartpho-word")
bartpho_word = AutoModel.from_pretrained("vinai/bartpho-word")
input_text = 'Chúng tôi là những nghiên cứu viên'
input_ids = tokenizer(input_text, return_tensors='pt')
features = bartpho_word(**input_ids)

Figure 1: An example code using BARTpho for feature extraction with transformers in Python. Here, a 7-syllable text sequence “Chúng tôi là những nghiên cứu viên”We are researchers forms a 4-word sequence “Chúng tôi là những nghiên cứu viên”.

under a comparable setting in terms of the relatively equal numbers of model parameters and pre-training data sizes [4, 6, 7]. BART is also used to pre-train monolingual models for other languages such as French [13] and Chinese [14].

3. Our BARTpho

This section describes the architecture, the pre-training data and the optimization setup, that we use for BARTpho.

3.1. Architecture

Both BARTpho_syllable and BARTpho_word use the “large” architecture with 12 encoder and decoder layers and pre-training scheme of BART [4]. In particular, pre-training BART has two stages: (i) corrupting the input text with an arbitrary noising function, and (ii) learning to reconstruct the original text, i.e. optimizing the cross-entropy between its decoder’s output and the original text. Here, BART uses the standard architecture Transformer [31], but employing the GeLU activation function [32] rather than ReLU and performing parameter initialization from $\mathcal{N}(0, 0.02)$. Following BART [4], we employ two types of noise in the noising function, including text in-filling and sentence permutation. For text in-filling, we sample a number of text spans with their lengths drawn from a Poisson distribution ($\lambda = 3.5$) and replace each span with a single special <mask> token. For sentence permutation, consecutive sentences are grouped to generate sentence blocks of 512 tokens, and sentences in each block are then shuffled in random order. Following mBART [10], we also add a layer-normalization layer on top of both the encoder and decoder.

3.2. Pre-training data

For BARTpho_syllable, we employ the PhoBERT pre-training corpus [15], that contains 20GB of uncompressed texts (about 145M automatically word-segmented sentences). In addition, we also reuse the PhoBERT’s tokenizer that applies a vocabulary of 64K subword types and BPE [17] to segment those word-segmented sentences with subword units. BARTpho_word has about 396M parameters. Pre-training data for BARTpho_word is a detokenized variant of the PhoBERT pre-training corpus (i.e. about 4B syllable tokens). We use the pre-trained SentencePiece model [18] from XLM-RoBERTa [33] used in mBART [10], to segment sentences with sub-syllable units and select a vocabulary of the top 40K most frequent types. BARTpho_syllable has about 396M parameters.

3.3. Optimization

We utilize the BART implementation with the denoising task from fairseq [19]. We use Adam [34] for optimization, and use a batch size of 512 sequence blocks across 8 A100 GPUs (40GB each) and a peak learning rate of 0.0001. Note that we initialize parameter weights of BARTpho_word by those from mBART. For each BARTpho model, we run for 15 training epochs in about 6 days (here, the learning rate is warmed up for 1.5 epochs).

3.4. Usage example

Figure 1 presents a basic usage of our pre-trained BARTpho models for feature extraction with transformers to show its potential use for other downstream tasks. More usage examples of BARTpho with both fairseq and transformers can be found at the BARTpho’s GitHub repository.

4. Experiments

4.1. Text summarization

We evaluate and compare the performance of BARTpho with the strong baseline mBART on a downstream generative task of Vietnamese text summarization. Here, mBART is pre-trained on a Common Crawl dataset of 25 languages, which includes 137 GB of syllable-level Vietnamese texts.

4.1.1. Experimental setup

We employ the single-document summarization dataset VNDS [35], consisting of 150704 news articles each including a news abstract (i.e. gold summary) and body content (i.e. input text). In particular, 105418, 22642 and 22644 articles are used for training, validation and test, respectively. However, we find that there are duplicate articles in this dataset. Therefore, we filter the duplicates, resulting in 99134, 22184 and 22498 articles for...
Table 1: Detokenized and case-sensitive ROUGE scores (in %) w.r.t. duplicate article removal. R-1, R-2 and R-L abbreviate ROUGE-1, ROUGE-2 and ROUGE-L, respectively. Every score difference between mBART and each BARTpho version is statistically significant with p-value < 0.05.

<table>
<thead>
<tr>
<th>Model</th>
<th>Validation set</th>
<th>Test set</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>R-1</td>
<td>R-2</td>
</tr>
<tr>
<td>mBART</td>
<td>60.06</td>
<td>28.69</td>
</tr>
<tr>
<td>BARTpho_{syllable}</td>
<td>60.29</td>
<td>29.07</td>
</tr>
<tr>
<td>BARTpho_{word}</td>
<td>60.55</td>
<td>29.89</td>
</tr>
</tbody>
</table>

Table 2: ROUGE scores (in %) w.r.t. the original dataset setting (i.e. without duplicate article removal). [...] denotes the best performing model among different models experimented from [35], and [*] denotes scores reported in [30].

<table>
<thead>
<tr>
<th>Model</th>
<th>Original validation set</th>
<th>Original test set</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>R-1</td>
<td>R-2</td>
</tr>
<tr>
<td>fastAbs [*]</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>viBERT2viBERT [*]</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>PhoBERT2PhoBERT [*]</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>mT5 [*]</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>mBART</td>
<td>60.39</td>
<td>29.19</td>
</tr>
<tr>
<td>BARTpho_{syllable}</td>
<td>60.89</td>
<td>29.98</td>
</tr>
<tr>
<td>BARTpho_{word}</td>
<td>61.10</td>
<td>30.34</td>
</tr>
</tbody>
</table>

Training, validation and test, respectively.\(^3\) When fine-tuning BARTpho_{syllable} and mBART, we use a detokenized version of the filtered dataset, while its automatically word-segmented version is used for fine-tuning BARTpho_{word}.

We formulate this task as a monolingual translation problem and fine-tune our BARTpho and the baseline mBART using the same hyper-parameter tuning strategy. We fix the maximum number of tokens in a batch at 4096. We use Adam and run for 20 training epochs. We also perform grid search to select the Adam initial learning rate from \{1e-5, 2e-5, 3e-5, 5e-5\}. We employ beam search with a beam size of 4 for decoding. We evaluate each model 4 times in every epoch. We select the model checkpoint that produces the highest ROUGE-L score [36] on the validation set, and we then apply the selected one to the test set. Note that we compute the detokenized and case-sensitive ROUGE scores for all models (here, we detokenize the fine-tuned BARTpho_{word}’s output before computing the scores).

4.1.2. Main results

Table 1 presents our obtained ROUGE scores on the validation and test sets for the baseline mBART and our two BARTpho versions w.r.t. the setting of duplicate article removal. Clearly, both BARTpho versions achieve significantly better ROUGE scores than mBART on both validation and test sets.

We also conduct a human-based manual comparison between the outputs produced by the baseline mBART and our two BARTpho versions. In particular, we randomly sample 100 input text examples from the test set; and for each input example, we anonymously shuffle the summary outputs from three fine-tuned models (here, each input sampled example satisfies that any two out of three summary outputs are not exactly the same). We then ask two external Vietnamese annotators to choose which summary they think is the best. We obtain a Cohen’s kappa coefficient at 0.61 for the inter-annotator agreement between the two annotators. Our second co-author then hosts and participates in a discussion session with the two annotators to resolve annotation conflicts (here, he does not know which model produces which summary). Table 1 shows final scores where our BARTpho obtains a better human evaluation result than mBART.

For comparison with previously published results [35, 30], we also fine-tune our BARTpho models and baseline mBART on the original training set (i.e. without duplicate article removal),\(^4\) using the same hyper-parameter tuning strategy as presented in Section 4.1.1. We report ROUGE scores on the original test set in Table 2. The previous best model from experiments in [35, 30] is PhoBERT2PhoBERT with a ROUGE-L score at 39.44. This score is 0.2 and 0.7 points lower than those of BARTpho_{syllable} and BARTpho_{word}, respectively. Tables 1 and 2 show that BARTpho helps attain a new SOTA performance for this task.

Our automatic and human evaluation results from tables 1 and 2 demonstrate the effectiveness of large-scale BART-based monolingual seq2seq models for Vietnamese. Note that mBART uses 137 / 20 \(\approx\) 7 times bigger Vietnamese pre-training data than BARTpho. In addition, the multilingual seq2seq mT5 [11] is pre-trained on the multilingual dataset mC4 that includes 79M Common Crawl Vietnamese pages consisting of 116B syllable tokens, i.e. mT5 uses 116 / 4 \(\approx\) 29 times bigger Vietnamese pre-training data than BARTpho. However, BARTpho surpasses both mBART and mT5, reconfiming that the dedicated language-specific model still performs better than the multilingual one [15]. Tables 1 and 2 also show that BARTpho_{word} out-performs BARTpho_{syllable}, thus demonstrating the positive influence of word segmentation for seq2seq pre-training and fine-tuning in Vietnamese.

\(^3\)Firstly, we remove duplicates inside each of the training, validation and test sets. Secondly, if an article appears in both training and validation/test sets, then the article is filtered out of the training set. Lastly, if an article appears in both validation and test sets, then the article is filtered out of the validation set.

\(^4\)This is not a proper experimental setup because of data leakage, e.g. 1466 training articles appear in the test set.
4.2. Capitalization and punctuation restoration

Most Automatic Speech Recognition (ASR) systems generate text transcripts without information about capitalization and punctuation, which limits the readability of the transcripts. In addition, using these lowercasing and non-punctuation types of ASR transcripts as input to downstream task models, e.g. named entity recognition, machine translation and the like, might also cause performance degradation [37] because the downstream task models are usually trained on well-formatted text datasets. Thus, capitalization and punctuation restoration are important steps in ASR transcript post-processing. An example enriching ASR transcripts with capitalization and punctuation restoration is as follows:

<table>
<thead>
<tr>
<th>A transcript</th>
</tr>
</thead>
<tbody>
<tr>
<td>chuỗi nhà hàng này gần đây đã phải đóng cửa một loạt các chi nhánh theo sở kế hoạch và đầu tư Hà Nội và thành phố Hồ Chí Minh, Golden gate đã đóng cửa bảy chi nhánh vào cuối năm 2015</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>The transcript enriched with capitalization and punctuation restoration &amp; its English translation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chuỗi nhà hàng này gần đây đã phải đóng cửa một loạt các chi nhánh. Theo Sở Kế hoạch và Đầu tư Hà Nội và Thành phố Hồ Chí Minh, Golden Gate đã đóng cửa bảy chi nhánh vào cuối năm 2015</td>
</tr>
</tbody>
</table>

Capitalization and punctuation restoration models generally fall into two main categories of approaches: sequence tagging [38, 39, 40] and sequence-to-sequence [41, 42]. In this investigation, we follow the sequence-to-sequence approach to evaluate and compare our BARTpho and mBART on the Vietnamese capitalization and punctuation restoration tasks. The models take lowercase, unpunctuated texts as input and produce true case, punctuated texts as output.

### 4.2.2. Main results

Table 3 presents the results obtained by our BARTpho and mBART on the capitalization task. We find that our BARTpho performs better than mBART. In particular, BARTpho syllable and BARTpho word, obtain 1.1% and 0.7% absolute higher F1 scores than mBART, respectively.

Table 3 also shows the obtained results of our BARTpho and mBART on the punctuation restoration task. Both BARTpho versions outperform mBART on the Comma and Question types, and the performance gap is substantial w.r.t. the latter mark. Furthermore, mBART does better than BARTpho on the Period mark, however, the performance gaps are small, i.e. mBART produces 0.14% and 0.4% higher scores than BARTpho word and BARTpho syllable, respectively. Overall, our BARTpho still outperforms mBART, where BARTpho word obtains the highest Overall F1 score.

### 5. Conclusion

In this paper, we have presented BARTpho syllable and BARTpho word—the first pre-trained and large-scale monolingual seq2seq models for Vietnamese. We demonstrate the usefulness of our BARTpho by showing that BARTpho performs better than its competitor mBART and helps produce the SOTA performance for the downstream generative task of Vietnamese text summarization. We also show that BARTpho is more effective than mBART on the Vietnamese capitalization and punctuation restoration tasks. We hope that our public BARTpho models can foster future research and applications of generative Vietnamese NLP tasks.

### 6. References


