On the Effectiveness of Quasi Character-Level Models for Machine Translation

Anonymous ACL submission

Abstract

Neural Machine Translation (NMT) models often use subword-level vocabularies to deal with rare or unknown words. Although some studies have shown the effectiveness of purely character-based models, these approaches have resulted in highly expensive models in computational terms. In this work, we explore the benefits of quasi-character-level models for low-resource NMT and their ability to mitigate the effects of the catastrophic forgetting problem. We first present a theoretical foundation along with an empirical study on the effectiveness of these models, as a function of the vocabulary and training set size, for a range of languages, domains, and architectures. Next, we study the ability of these models to mitigate the effects of catastrophic forgetting in machine translation. Our work suggests that quasi-character-level models have practically the same generalization capabilities as character-based models but at lower computational costs. Furthermore, they appear to help achieve greater consistency between domains than standard subword-level models, although the catastrophic forgetting problem is not mitigated.

1 Introduction

Neural machine translation (NMT) has become the dominant paradigm in the field of machine translation due to the impressive results achieved with encoder-decoder architectures (Sutskever et al. (2014); Cho et al. (2014); Wu et al. (2016); Vaswani et al. (2017)).

Despite these advances, the translation of rare or unknown words became a more complex problem than initially thought. Consequently, authors proposed different approaches that can be grouped into three categories: i) Character-based models ii) Hybrid NMT models. iii) Subword-level models.

Character-based models can naturally deal with rare or unseen words as they contain the minimum set of characters to build all the words in a language. However, these models have historically resulted in unsatisfactory results (Vilar et al. (2007); Neubig et al. (2013)) or highly expensive models in computational terms (Luong and Manning, 2016a).

Later, Hybrid NMT models appeared to close the gap between word- and character-based representations (Luong and Manning, 2016b). The idea behind these models is to translate mainly at the word level and only query character components for rare words when necessary. However, these models tend to be a bit cumbersome due to the need for two models to do the back-off. Finally, word segmentation approaches such as BPE (Sennrich et al., 2016), or Unigram (Kudo, 2018) emerged to encode words using a vocabulary of subwords units efficiently.

Despite the success of subword-level models and the evidence that each data set has an optimal vocabulary size (Gowda and May, 2020), there is no clear way to determine this optimal size without resorting to trial and error. However, it has been known that character-level models tend to work better for extremely low resource settings.

Some researchers might argue that with the increase of data volume and mining techniques, low-resources languages are no longer a problem in NMT. However, this is not entirely true since many languages are spoken but not written on the internet (e.g., Tigrinya, Sotho, Tsonga, etc.).

Motivated by these ideas, we decided to study whether NMT quasi character-based models had the same advantage as character-based approaches for low-resource scenarios but at much lower computational costs due to the exponential decrease in the average number of tokens per sentence when highly frequent char-pairs are merged.

Furthermore, we decided to study if these models could mitigate the effects of the catastrophic forgetting phenomenon by exploiting its vocabulary similarity between domains.

The contributions of this paper are twofold:

- Quasi-character-level models appear to outperform their character-based in terms of performance while practically offering the same generalization capabilities at much lower computational costs.

- Quasi-character-level models appear to achieve higher consistencies between domains, although they also seem to be more susceptible to the effects of catastrophic forgetting.

2 Related work

Character-based models have been well-studied in the Natural Language Processing (NLP) field to deal with the open-vocabulary problem. One of the first character-based models was proposed by Vilar et al. (2007), who treated the source and target sentences as a string of letters. Similarly, Neubig et al. (2013) viewed translation as a single transduction between character strings in the source. However, their results were not satisfactory as their models generally performed worse than their word-based counterparts.

Consequently, authors proposed multiple strategies based on Hybrid NMT models (Luong and Manning, 2016b) and subword-level representations (Sennrich et al. (2016); Kudo (2018)) to get the best of both worlds.

Luong and Manning (2016a) and Costa-jussà and Fonollosa (2016) showed that competitive purely character-based NMT models were possible but extremely slow to train and infer. Chung et al. (2016) demonstrated that an NMT model with a character-based decoder could outperform NMT models with subword-level decoders.

Many authors have studied the Zipfian nature of languages in NMT. For instance, Gowda and May (2020) did it to find the optimal vocabulary size, and Raunak et al. (2020) to characterize the long-tailed phenomena in NMT. Similarly, Cherry et al. (2018) showed that character-level models have their greatest advantage when data sizes are small, and Sennrich and Zhang (2019) that reducing vocabulary size improves low-resource NMT.

Finally, this paper ends with a brief discussion on the ability of quasi-character-based models to mitigate the catastrophic forgetting problem in NMT. As far as we know, this is the first work that addresses this problem from this perspective, since most of the works that we know of are based on regularization (Li and Hoiem (2016) and Kirkpatrick et al. (2016)), dynamic architectures (Rusu et al. (2016) and Draelos et al. (2016)) or Complementary Learning Systems (CLS) (Kemker and Kanan (2017)).

3 Neural Machine Translation

3.1 Neural architectures for Machine Translation

The goal of any translation system is to transform an input sequence in a given language into an output sequence in a target language.

Nowadays, this is usually done using neural models based on the encoder-decoder architecture, also known as seq2seq models in the machine translation community ((Sutskever et al., 2014)). The encoder part transforms the input sequence into an internal representation, and then the decoder transforms this internal representation into the output sequence.

Recurrent architectures (RNNs) were the first to be successfully applied in an encoder-decoder setup for machine translation. Even though there are many RNNs, most of them chain a series of unit cells sequentially to process temporal sequences. We decided to use LSTMs ((Hochreiter and Schmidhuber, 1997)) because their units cells are explicitly designed to deal with long-term dependencies.

Convolution-based architectures (CNN) do not contain any recurrent elements. They can do this because the idea behind this architecture is that the convolutional filters can slide through the sequence of tokens from beginning to end ((Gehring et al., 2017)).

Lastly, Vaswani et al. (2017) introduced the Transformer architecture, which is a state-of-the-art model based entirely on the concept of *attention* (Bahdanau et al. (2015); Luong et al. (2015)) to draw global dependencies between the input and output. Unlike RNNs or CNNs, this architecture processes its temporal sequences all at once through the use of masks that encode their temporal information.

This work is focused on the Transformer as it is the current state-of-the-art model for NMT. Nonetheless, RNNs and CNNs are briefly explored for completeness.

3.2 The open vocabulary problem

In the written language, it is common to find alternative spellings (i.e., *color-colour*) and typos (i.e., *acknowledge-acknowlege*) that slightly modify the spelling of a word but do not prevent us, the humans, from understanding its meaning. However, suppose a model is using a word-level representation. In that case, it will stop knowing a *known word* at the very first moment that it is slightly modified (and this modification is not in its vocabulary). Similarly, it has to be taken into account that many languages use agglutination and compounding mechanisms to form new words, making word-based vocabularies a very inefficient strategy.

As a result, researchers have proposed multiple approaches to deal with the open vocabulary problem. These approaches can be mostly grouped into three categories: i) Character-based models, ii) Hybrid NMT models iii) Subword-level models.

Character-based models contain the minimum possible vocabulary to form all possible words in a language. Therefore, these models can translate rare or even unseen words character-by-character, but at the same time, these models tend to be much slower and harder to train than word-based models, as they have to deal with longer long-term dependencies.

Hybrid NMT approaches can be seen as a "trick", as they translate primarily at word-level but fall back to character-level when a rare or unseen word appears.

Lastly, subword-level representations allow us to efficiently represent words as a sequence of subwords units. Although they practically solved the *unknown* problem of word-based approaches, they cannot solve it completely. To do so, the current approach is to perform *byte-fallback*.

A side effect of subword-level representations is that by changing the vocabulary size limit, they can (partially) degenerate to character- or word-based representations, which allow us to study the effects of the vocabulary more closely.

4 Experimental setup

4.1 Datasets

The data used for this work comes mainly from the WMT tasks (see Table 1)

These datasets contain parallel sentences from different languages and domains (political, economic, health, biological, talks, etc.).

In addition to the original datasets, we have created smaller versions[1] for some training sets in order to study the effects of the vocabulary size as a function of training size (from low- to high-resource language).

[1]The validation and test sets were shared across training set versions

Dataset	Training set
Europarl (es-en)	1.9M/100K/50K
Europarl (de-en)	1.8M/100K/50K
Europarl (cs-en)	635K/100K/50K
CommonCrawl (es-en)	1.8M/100K
SciELO (es-en)	575K/120K
NewsCommentary (de-en)	357K/35K
Tatoeba (mr-en; mk-en)	50K
IWLST'16 (de-en)	196K
Multi30K (de-en)	29K

Table 1: All the values in this Table indicate the number of sentences.

4.2 Training details

All the preprocessing was done using SentencePiece (Kudo and Richardson, 2018), with Unigram (Kudo, 2018) as the tokenization model.

To train our models[2] we used Fairseq v1.0.0a0(Ott et al., 2019), with a pretty standard set of training hyper-parameters[3]. We tried to use similar settings on most models. However, we noticed that as we use smaller vocabularies and training sets, these models became more sensitive to the given hyper-parameters. This was particularly true on character-based models.[4]

Similarly, we also experimented with different neural architectures (Transformer, LSTMs, and CNNs). In the case of the Transformer, we began to experiment with the *Standard Transformer* (45-93M parameters), but then we switched to a smaller version (4-25M parameters), as both performed quite similarly in terms of performance (±1 BLEU), and the later was notably faster.

4.3 Evaluation metrics

We evaluate all models using Sacrebleu (Post, 2018), which produces shareable, comparable, and reproducible BLEU scores (Papineni et al., 2002). Similarly, we also made use of BERTScore (Zhang et al., 2019), a state-of-the-art neural metric for machine translation.

[2]We use 2x NVIDIA GP102 (TITAN XP) - 12GB

[3]Hyper-parameters: *lr=[0.5e-4, 1e-3], weight-decay=[1e-3, 1e-4], criterion=[ce, label-ce(0.1)], scheduler=[fixed, inverse-sqrt], warmup-updates=[4000], optimizer=[adam, sgd, nag], clip-norm=[0.0, 0.1, 1.0], beam-width=5]*

[4]Most trainings last between a few hours to 1-2 days

5 Experimentation

5.1 On Quasi-Character-Level Hypotheses

We will say that a vocabulary is complete, if and only if we can represent any possible word of a given language. Therefore, given two different vocabularies, A and B, we will say that they are grammatically equivalent if they are complete. Similarly, the smaller a vocabulary is, the greater the generalization capability of the model used will have to be, as the amount of information per token will be diluted by the number of tokens needed to encode each string.

Based on these premises, we can infer that the representation power of a given model will depend on the degree of generalization required by its vocabulary, the amount of data required to learn it and if the complexity of the model can handle it.

In practice, this means that, given a model with enough complexity, the generalization advantage of character-based vocabularies with respect to subword-based or word-based vocabularies decreases as the amount of data increases.

From these theories, supported by empirical evidence (Sennrich and Zhang, 2019), we hypothesize that quasi-character-based models should perform similar to character-based models in low-resource environments but with much lower computational costs.

It is essential to highlight that a quasi-character-level vocabulary is meant to depict a subword-level vocabulary which is an order or two orders of magnitude smaller than standard subword-level vocabularies. The motivation for these vocabularies is to provide practically the same generalization capabilities as character-level models, but more efficiently, by exploiting highly frequent n-grams to decrease the sentence length exponentially.

5.2 Effects of the vocabulary and corpus size

In order to test the basis of our hypothesis, we chose a medium-sized corpus such as Europarl-2M (de-en). Then, we created two other versions, where the training set was artificially reduced from 2M sentences to 100k and 50k sentences. Similarly, we created two vocabularies:

- A standard subword-level vocabulary with 32k entries
- A quasi-character-level vocabulary with 350 entries

The aim of this experiment was twofold. First, we sought to ratify the observations made by other authors that smaller vocabularies tend to help in low-resource environments (Cherry et al. (2018)). However, in our case, we provide additional data points for smaller datasets (less than 2M sentences), languages, and domains (following sections). Second, we sought to establish baselines for our quasi-character-based models so that we could later study their computational advantage over purely character-level models.

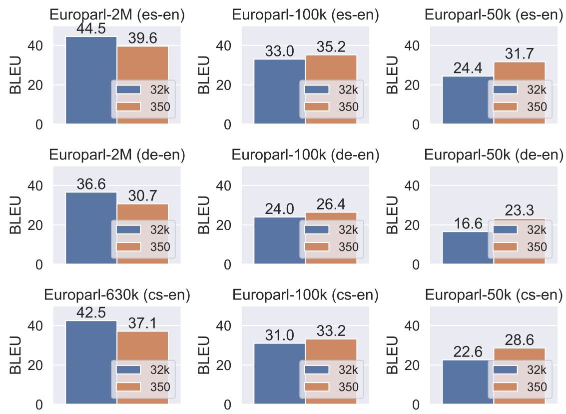


Figure 1: As we limit the training data (left to right), we see that quasi character-level Transformers performs better than their large subword vocabularies versions. Similarly, this phenomenon seems to occur regardless of the language (top to bottom).

As expected, in Figure 1 we see that when there is enough training data, standard subword-level models outperform models with quasi-character-level vocabularies (first column). In contrast, as the amount of training data is reduced (second and third columns), quasi-character-level models outperformed the standard subword-level models.

In total, we performed this experiment for three different language pairs (Spanish-English, German-English, and Czech-English) in order to account for potential language biases and domains (political, economical, health, biological, transcripted talks, etc.), to be able to generalize the findings of previous authors to much smaller corpus and especially, to quasi-character-level vocabularies (See section 5.4).

5.3 On the Effectiveness of Quasi-Character-Level Models

As other studies have shown (Gowda and May, 2020), each dataset seems to have an optimal vocabulary size. Therefore, this could imply that the results from our previous experiment could be biased towards a sub-optimal vocabulary size. To account for these possible biases, we performed a similar set of experiments in which we gradually increased the vocabulary size (at the subword-level) from 100 entries to 16,000 entries (plus 256 additional entries for the byte-fallback) in order to obtain the characteristic curve for each dataset as a function of the vocabulary size (See Figure 2).

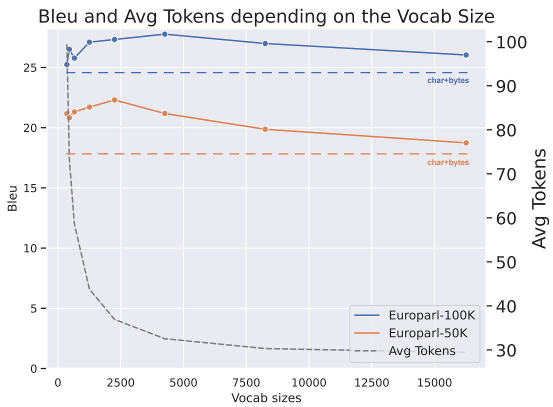


Figure 2: As we decrease the size of the vocabulary, the average number of tokens per sentence increases exponentially, so more complex models and more training data are needed to exploit the additional generalization capabilities of these vocabularies. However, by merging a few highly frequent char-pairs into a single token, we can have models that practically generalize as character-based models but with much lower computational costs at training and inference.

In Figure 2 we can see two variations of the Europarl (de-en) dataset to emulate low-resource settings, one with 50k sentences (orange line) and another with 100k sentences (blue line).

The first thing to notice in this Figure 2 is that, as we limit the number of entries in the vocabulary in resource-poor environments, the performance of our models increases. Although this was expected, it was necessary to add more robustness to our previous conclusions. Similarly, it is also important to point out that as we increase the amount of training data (Europarl-100k), the phenomenon described here is still present. Nonetheless, it is less noticeable than in the smaller corpus (Europarl-50k), as expected. This observation might indicate that for high-quality corpus, the advantage of character-level models could disappear much quicker than was previously thought (Cherry et al., 2018).

Then, we see that as vocabularies approach character-level representations, the average number of tokens per sentences increases exponentially (dashed line), which directly impacts the performance of the models due to: i) The additional complexity needed to handle the greater generalization capabilities of smaller vocabularies. ii) The problems imposed by having to deal with longer long-term dependencies. iii) Higher computational costs at training and run-time.

However, as we can see in Figure 2 when we increase the vocabulary size, the average number of tokens per sentence decreases exponentially. The direct consequence of this is that quasi-character-level models outperformed purely character-based models (dashed lines) by a significant margin without increasing the complexity of this model or the training time. Following these observations, we wonder what may be the benefit of using character-based models instead of quasi-character-based models, since a slight increase in vocabulary size leads to the collapse of highly frequent pairs that individually contribute little to the model's learning, but when collapsed, produce considerable reductions in average sentence size, which results in much lower computational costs and fewer problems from learning long-term dependencies and complex generalizations.

5.4 On the Generalization of Quasi-Character-based approaches

In this section, we study if the benefits of Quasi-Character-based approaches generalize to other languages, domains, and neural architectures.

5.4.1 Domain generalization

As we have briefly described in Section 5.2, seemed to generalize to other Latin-based languages such as Spanish, German and Czech. However, we wondered whether the domain could be introducing some biases since the Europarl dataset only contains parallel sentences extracted from the European Parliament website.

To do so, we repeated the experiment but on parallel corpus from different domains, such as crawled data (CommonCrawl), political and economic news (NewsCommentary), health and biological sciences (SciELO), transcripted talks (IWLST'16), and multimodal transcriptions (Multi30k).

Interestingly, Quasi-Character-Level models kept outperforming their standard subword-level counterparts when then the training data was artificially reduced to emulate low-resource environments, so it seems that the advantages of quasi-character-level models seem to be present regardless of the domain (See Figure 3).[5]

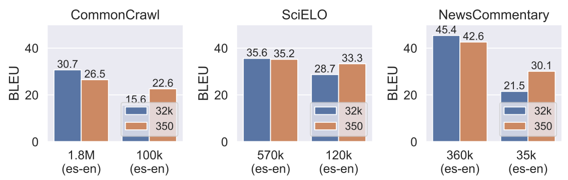


Figure 3: The benefits of quasi-character-level models for low-resource settings appear to be consistent regardless of the domain.

5.4.2 Non-Latin and Low-Resource Languages

After that, we wanted to study this phenomenon for non-Latin languages and actual low-resource languages. To do so, we use the Tatoeba dataset for Marathi and Macedonian, both with 50K sentences (See Figure 4).

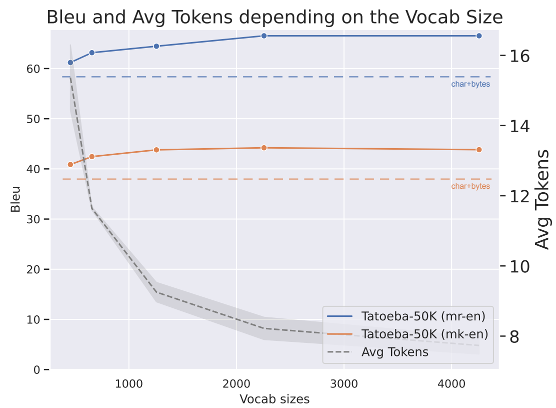


Figure 4: Quasi-Character-Level models outperformed Character-based models for Marathi and Macedonian (Tatoeba), using half of the average tokens per sentence. A non-negligible optimization due to the quadratic complexity of the Transformer's self-attention

Again, in Figure 4 we see that Quasi-Character-Level models outperformed their character-based counterparts. However, this time we could not compare the difference against standard subword-level vocabularies (8k, 16k, and 32k) because there was too little training data to build those vocabularies[6].

Nonetheless, the important thing to highlight here is that when we use a vocabulary of around 1000 entries, the average number of tokens per sentence was half of the character-based model, which is non-trivial in computational- and memory terms due to the quadratic complexity of the self-attention of the Transformer architecture.

5.4.3 Neural architecture generalization

In this section, we were interested in studying how much of the advantage of quasi-character-level models was due to the ability of the Transformer architecture to learn long-term dependencies. Therefore, we briefly study if our findings could generalize to other neural machine translation architectures, such as LSTMs or CNNs. Specifically, we focused our work on bidirectional LSTMs with attention mechanisms and fully convolutional architectures like the one described in (Gehring et al., 2017).

Although the comparison of different neural architectures is not trivial, we tried to naively explore this topic by only comparing models that had a similar number of parameters for a given vocabulary (i.e., 25-30M parameters for vocabularies of 32k subwords.

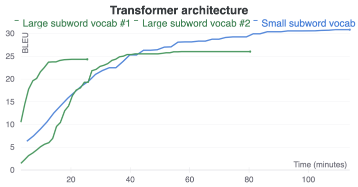
From our experimentation, we observed that when standard subword-level models were trained with enough data (all available data), they outperformed all character- and quasi-character-level models regardless of their architecture. However, when this experiment was repeated on the low-resource regime, the quasi-character-based models performed better than their standard subword-level counterparts when Transformer or Bi-LSTM architectures were used. Furthermore, if CNNs were had given more training time[7], it is highly likely that they would have outperformed the standard subword-level models too (see Figure 5).

In the left figure 5a, we see that quasi-character-level Transformers consistently outperform the ones with standard subword-level vocabularies. This phenomenon is still present for LSTMs (the central Figure 5b). However, it is not as evident as with the Transformer architecture due to the problems of RNNs with modeling long-term dependencies. Finally, we see in the right Figure 5c

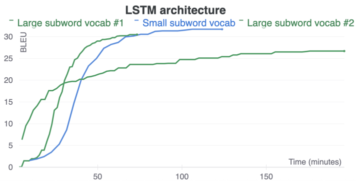
[5] We have considered the results from IWLST'16 and Multi30K redundant, so we decided not to included a Figure for them. Nonetheless, Quasi-Character-Level models improved the BLEU score in 6.2pts for the IWLST'16 dataset and 2.3pts for the Multi30k dataset.

[6] We could have used the number of merge operations instead of the vocabulary size, but since it is not really a fair comparison, we decided to make the comparison amongst small vocabularies.

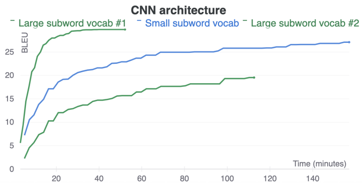
[7] The max-epochs hyper-parameter stopped the training, and due to the lack of time we had not been able to repeat it



(a) Transformer architecture



(b) Bi-LSTM architecture



(c) CNN architecture

Figure 5: The green lines refer to the best and worst runs of the models with standard subword-level vocabularies, while the blue lines refer to the best run of the quasi character-level models.

that CNNs cannot easily model long-term dependencies, so they do not benefit as easily from the quasi-character-level representations

From these results, we conclude that the ability of a neural architecture to model long-term dependencies is critical to obtain significant benefits from character- or quasi-character-based approaches.

5.5 On the Catastrophic Forgetting Problem

In this section, we study whether quasi-character-level models could help mitigate the effects of the catastrophic forgetting problem, whereby neural networks forget previously learned information after learning new information.

To do this, we designed an experiment in which we first train a model in a domain A and evaluate it in domains A and B to establish the baselines. Next, we fine-tune the model trained in domain A with data from the new domain B, and then, it is evaluate it in domains A and B. In theory, the model trained in domain A should perform well in the domain A, and poorly in the unseen domain B. Similarly, after the fine-tuning on domain B, it should perform worse in A and better in domain B than the original model trained only on domain A.

In Figure 6a we see that the quasi-character-level model trained on the health domain (SciELO) obtained a BLEU of 33.3pts on its domain (Health) and a BLEU of 14.3pts in the other domain (Biological). Then, when we fine-tune it on the Biological domain (SciELO), the BLEU obtained on this domain increased from 14.3 to 31.7pts, while BLEU for the health domain fell from 33.3 to 21.0pts. In Figure6b we see that something similar happened for the standard subword-vocabulary. However, the effects of the catastrophic forgetting problem were not as strong as in the other model because the BLEU score went from 28.7 to 28.0pts.

From Figure 6, we can infer that the vocabulary seems to have a substantial impact on the effects of catastrophic forgetting because character-level

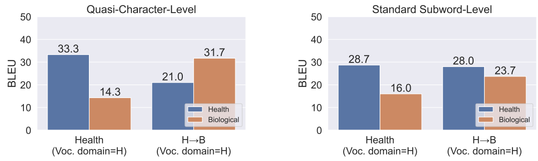


Figure 6: Vocabularies seem to have a strong impact on the catastrophic forgetting effects. While the quasi character-level model lost 12.3pts, the large subword-level model only lost 0.7pts

vocabularies seem to make models more susceptible to the catastrophic forgetting problem than standard subword-level vocabularies.

To further explore this problem, we repeated the previous experiment but taking into account the vocabulary domain. As a result, we discovered that the vocabulary domain has a stronger impact on the model's performance than we thought. As shown in Figure 7, quasi-character-level models seem to be highly consistent between domains, while standard subword-level models seem to be particularly sensitive to their vocabulary's domain, to the point of achieving opposite results between domains (see right column of the Figure 7).

Even though quasi-character-level models achieved better consistencies across domains, they appear to suffer more severely from the effects of the catastrophic forgetting problem than their standard subword-level counterparts. We believe that by using specially designed regularization techniques to address this issue, such as LwF ((Li and Hoiem, 2016)) or EWC ((Kirkpatrick et al., 2016)) these problems could be mitigated, leading to more robust and consistent models.

6 Conclusion

In this paper, we have empirically studied the effectiveness of quasi-character-level models in terms of performance and computational efficiency

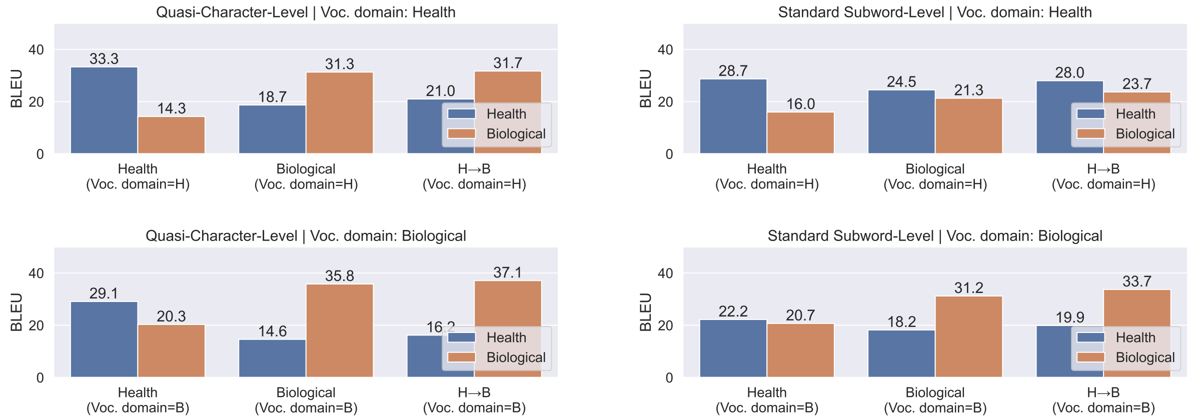


Figure 7: Quasi-character-level models (left figures) appear to be more consistent between domains than models with standard subword-level vocabularies (right figures)

with regard to purely character-based and standard subword-level models. In addition to this, we have studied the generalization of quasi-character-level vocabularies and their ability to tackle the catastrophic forgetting problem.

Our studies reveal that quasi-character-level models offer virtually the same generalization capabilities as character-level models but with much lower computational costs. Similarly, these models outperformed character-based and standard subword-level models on low-resource settings for a wide range of languages, domains, and neural architectures.

Finally, we have showed that even though quasi-character-level vocabularies do not seem to mitigate the effects of the catastrophic forgetting problem, they achieved a higher consistencies between domains, which could lead to substantial improvements if specific regularization techniques are applied to deal with catastrophic forgetting.

Acknowledgment

Work supported by the Horizon 2020 - European Commission (H2020) under the SELENE project (grant agreement no 871467) and the project Deep learning for adaptive and multimodal interaction in pattern recognition (DeepPattern) (grant agreement PROMETEO/2019/121). We gratefully acknowledge the support of NVIDIA Corporation with the donation of a GPU used for part of this research.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473.

Colin Cherry, George F. Foster, Ankur Bapna, Orhan Firat, and Wolfgang Macherey. 2018. Revisiting character-based neural machine translation with capacity and compression. *CoRR*, abs/1808.09943.

Kyunghyun Cho, Bart van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the properties of neural machine translation: Encoder–decoder approaches. In *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, pages 103–111. Association for Computational Linguistics.

Junyoung Chung, Kyunghyun Cho, and Yoshua Bengio. 2016. A character-level decoder without explicit segmentation for neural machine translation. *CoRR*, abs/1603.06147.

Marta R. Costa-jussà and José A. R. Fonollosa. 2016. Character-based neural machine translation. In *Proceedings of the 54th Annual Meeting of the ACL (Volume 2: Short Papers)*, pages 357–361.

Timothy J. Draelos, Nadine E. Miner, Christopher C. Lamb, Craig M. Vineyard, Kristofor D. Carlson, Conrad D. James, and James B. Aimone. 2016. Neurogenesis deep learning. *CoRR*, abs/1612.03770.

Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N. Dauphin. 2017. Convolutional sequence to sequence learning. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 1243–1252.

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

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9:1735–80.

Ronald Kemker and Christopher Kanan. 2017. Fearnet: Brain-inspired model for incremental learning. *CoRR*, abs/1711.10563.

James Kirkpatrick, Razvan Pascanu, Neil C. Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. 2016. Overcoming catastrophic forgetting in neural networks. *CoRR*, abs/1612.00796.

Taku Kudo. 2018. Subword regularization: Improving neural network translation models with multiple subword candidates. In *Proceedings of the 56th Annual Meeting of the ACL (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, EMNLP 2018: System Demonstrations, Brussels, Belgium, October 31 - November 4, 2018*, pages 66–71.

Zhizhong Li and Derek Hoiem. 2016. Learning without forgetting. *CoRR*, abs/1606.09282.

Minh-Thang Luong and Christopher D. Manning. 2016a. Achieving open vocabulary neural machine translation with hybrid word-character models. In *Proceedings of the 54th Annual Meeting of the ACL (Volume 1: Long Papers)*, pages 1054–1063.

Minh-Thang Luong and Christopher D. Manning. 2016b. Achieving open vocabulary neural machine translation with hybrid word-character models. In *Proceedings of the 54th Annual Meeting of the ACL (Volume 1: Long Papers)*, pages 1054–1063.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on EMNLP*, pages 1412–1421.

Graham Neubig, Taro Watanabe, Shinsuke Mori, and Tatsuya Kawahara. 2013. Substring-based machine translation. *Machine Translation*, 27(2):139–166.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. *CoRR*, abs/1904.01038.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on ACL*, ACL '02, page 311–318.

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.

Vikas Raunak, Siddharth Dalmia, Vivek Gupta, and Florian Metze. 2020. On long-tailed phenomena in neural machine translation. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3088–3095, Online. Association for Computational Linguistics.

Andrei A. Rusu, Neil C. Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. 2016. Progressive neural networks. *CoRR*, abs/1606.04671.

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 ACL (Volume 1: Long Papers)*, pages 1715–1725.

Rico Sennrich and Biao Zhang. 2019. Revisiting low-resource neural machine translation: A case study. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 211–221, Florence, Italy. Association for Computational Linguistics.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *NIPS*, volume 27.

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 *Proceedings of the 31st NeurIPS*, NIPS'17, page 6000–6010.

David Vilar, Jan-T. Peter, and Hermann Ney. 2007. Can we translate letters? In *Proceedings of the Second WMT*, StatMT '07, page 33–39.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Lukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *CoRR*, abs/1609.08144.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with BERT. *CoRR*, abs/1904.09675.