Neural Machine Translation by Generating Multiple Linguistic Factors

Mercedes García-Martínez(✉), Loïc Barrault, and Fethi Bougares

LIUM, Le Mans University, Le Mans, France
{mercedes.garcia_martinez,Loic.Barrault,Fethi.Bougares}@univ-lemans.fr

Abstract. Factored neural machine translation (FNMT) is founded on the idea of using the morphological and grammatical decomposition of the words (factors) at the output side of the neural network. This architecture addresses two well-known problems occurring in MT, namely the size of target language vocabulary and the number of unknown tokens produced in the translation. FNMT system is designed to manage larger vocabulary and reduce the training time (for systems with equivalent target language vocabulary size). Moreover, we can produce grammatically correct words that are not part of the vocabulary. FNMT model is evaluated on IWSLT'15 English to French task and compared to the baseline word-based and BPE-based NMT systems. Promising qualitative and quantitative results (in terms of BLEU and METEOR) are reported.

Keywords: Machine translation · Neural networks · Deep learning · Factored representation

1 Introduction and Related Works

In contrast to the traditional phrased-based statistical machine translation [12] that automatically translates subparts of the sentences, standard Neural Machine Translation (NMT) systems use the sequence to sequence approach at word level and consider the entire input sentence as a unit for translation [2,5,25].

Recently, NMT showed better accuracy than existing phrase-based systems for several language pairs. Despite these positive results, NMT systems still face several challenges. These challenges include the high computational complexity of the softmax function which is linear to the target language vocabulary size (Eq. 1).

$$p_i = e^{o_i} / \sum_{r=1}^{N} e^{o_r} \text{ for } i \in \{1, \ldots, N\} \tag{1}$$

where o_i are the outputs, p_i their softmax normalization and N the total number of outputs.

In order to solve this issue, a standard technique is to define a *short-list* limited to the s most frequent words where $s << N$. The major drawback of this technique is the growing rate of unknown tokens generated at the output. Another work around has been proposed in [11] by carefully organising the

N. Camelin et al. (Eds.): SLSP 2017, LNAI 10583, pp. 21–31, 2017.
DOI: 10.1007/978-3-319-68456-7_2

batches so that only a subset K of the target vocabulary is possibly generated at training time. This allows the system to train a model with much larger target vocabulary without substantially increasing the computational complexity. Another possibility is to define a structured output layer (SOUL) to handle the words not appearing in the shortlist. This allows the system to always apply the softmax normalization on a layer with reduced size [14]. The problem of unknown words was addressed making use of the alignments produced by an unsupervised aligner [16]. The unknown generated words are substituted in a post-process step by the translation of their corresponding aligned source word or copying the source word if no translation is found. The translation of the source word is made by means of a dictionary.

Other recent work have used subword units instead of words. In [24], some unknown and rare words are encoded as subword units with the Byte Pair Encoding (BPE) method. Authors show that this can also generates words unseen at training time. As an extreme case, the character-level neural machine translation has been presented in several works [6,7,15] and showed very promising results. The character-level NMT architectures are composed of many layers, to deal with the long distance dependencies, increasing aggressively the computational complexity of the training process. In [22] has been shown that character-level decoders outperform subwords units using BPE method when processing unknown words, but they perform worse when extracting morphosyntactic information about the sentences, due to the long distances.

Among other previous works, our work can be seen as a continuation of [9]. Several works have used factors as additional information for the input words in neural language modelling with interesting results [1,18,26]. More recently, factors have also been integrated into a word-level NMT system as additional linguistic input features [23]. Unlike these previous works, we are considering factors as translation unit. We refer to *factors* as some linguistic annotations at word level, *e.g.* the Part of Speech (POS) tag, number, gender, etc. The advantages of using factors as translation unit are two-fold: reducing the output vocabulary size and allowing to generate surface forms which are never seen in the training data.

Factors were first introduced for NMT at output side in [9] where two factored synchronous symbols are simultaneously generated. Authors presented an investigation of the architecture of their factored NMT system to show that better results are obtained using a feedback of the two generated outputs concatenation.

Our work is different from previous efforts in that we consider only the best type of feedback for the network. We also introduce an additional factor about the case information (lowercase, uppercase or in capitals) and evaluate using a different translation test. Moreover, we apply an unknown words (*unk*) replacement technique using the alignments of the attention mechanism to replace the generated unknown words in target side. For that, we make use of an unigram dictionary to find the translation of the source word corresponding to the generated *unk*.

We compare this architecture to the state of the art BPE approach and the classic word-level NMT approach on the English to French dataset from IWSLT'15 evaluation campaign. We provide, in addition a quantitative and qualitative study about the obtained results.

The remainder of this paper is organized as follows: Sect. 2 describes the attention-based NMT system and Sect. 3 its extension using the factored approach. In Sect. 4, we describe the experiments and the obtained results. Finally, Sect. 5 concludes the paper and presents the future work.

2 Neural Machine Translation

The standard NMT model consists of a sequence to sequence encoder-decoder of two recurrent neural networks (RNN), one used by the encoder and the other by the decoder. The source language sequence is mapped into an embedded dimension in the encoder and the decoder maps the representation back to a target language sequence.

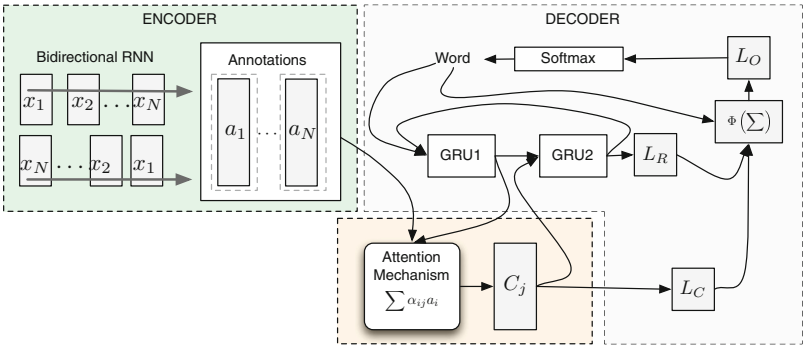

Fig. 1. Attention-based NMT system (Color figure online)

The architecture includes a bidirectional RNN encoder (see left part of Fig. 1) equipped with an attention mechanism [2]. Each input sentence word x_i ($i \in 1 \dots N$ with N the source sequence length) is encoded into an annotation a_i by concatenating the hidden states of a forward and a backward RNN provided by a gated recurrent unit (GRU) [5] to control the flow of information. These annotations $a_1 \dots a_N$ represents the whole sentence with a focus on the word being processed. One difference from the architecture of [2] is that the decoder contains a conditional GRU [8] which consists of two GRUs interspersed with the attention mechanism (see right top part of the Fig. 1). The first GRU combines the embedding of the previous decoded token and the previous hidden state in order to generate an intermediate representation which is an input of the attention mechanism and the second GRU. The attention mechanism (bottom yellow

part of the Fig. 1) computes a source context vector C_j as a convex combination of annotation vectors, where the weights of each annotation are computed locally using a feed-forward network. These weights can be used to align the target words with the source positions. The second GRU generates the hidden state of the conditional GRU by looking at the output of the first GRU and the context vector C_j. The decoder RNN takes as input the embedding of the previous output word (feedback of the network) in the first GRU, the context vector C_j in the second GRU and its hidden state. The output layer L_O is connected to the network through a hyperbolic tangent sum operation $\Phi(\sum)$ which takes as input the embedding of the previous output word as well as the context vector and the output of the decoder from the second GRU (both adapted with a linear transformation, respectively, L_C and L_R). Finally, the output probabilities for each word in the target vocabulary are computed with a *softmax* function. The word with the highest probability is the translation output at each timestep. The encoder and the decoder are trained jointly to maximize the conditional probability of the reference translation.

3 Factored Neural Machine Translation

The Factored Neural Machine Translation (FNMT) [9] is an extension of the standard NMT architecture which allows the system to generate several output symbols at the same time.

For the sake of simplicity, only two symbols are generated: the lemma and the concatenation of the different factors (verb, tense, person, gender, number and case information). The target words are then represented by a factored output: lemmas and factors. Factors may help the translation process providing grammatical information to enrich the output. The task of this work is English to French translation, English is a grammatically poor language and factors do not help for its translation, this has been tested in previous experiments. Therefore, we apply the factors only in the target side when translating to French which is a grammatically rich language. In the example shown in Fig. 3, from the verbal form in French *devient*, we obtain the lemma *devenir* and its factors *VP3#SL* (**V**erb, in **P**resent, **3**rd person, no gender (#), **S**ingular and **L**owercased form). Moreover, we can see the word *interéssant* with the lemma *interéssant* and factors *Adj##MSL* (**Adj**ective, no tense (#) and no person (#), **M**asculine gender, **S**ingular number and **L**owercased form). The morphological analyser MACAON toolkit [17] is used to obtain the lemma and factors for each word taking into account its context with nearly 100% accuracy. The first entry is used in the few cases that MACAON proposes multiple words (e.g. same word written in two forms).

The FNMT architecture is presented in Fig. 2. The encoder and attention mechanism of Fig. 1 remain unchanged. However, the decoder has been modified to get multiple outputs. The hidden state of the conditional GRU (cGRU) is shared to produce simultaneously several outputs. The output from the layer L_O has been diversified to two *softmax* layers, one to generate the lemma and

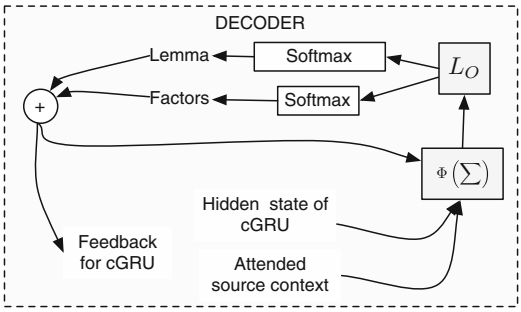

Fig. 2. Detailed view of the decoder of the Factored NMT system

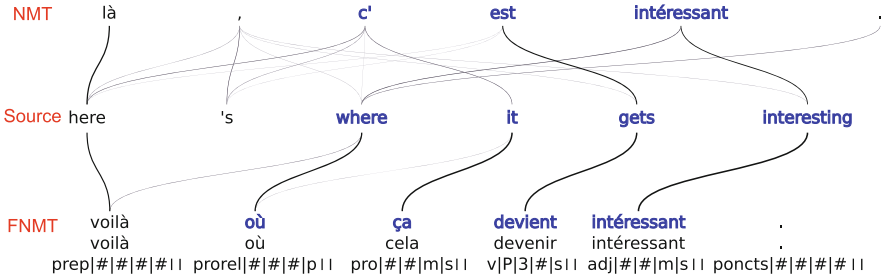

Fig. 3. Examples of NMT and FNMT outputs aligned against the source sentence

the other to generate the factors. An additional design decision is related to the decoder feedback. Contrary to the word based model, where the feedback is naturally the previous word (see Fig. 1), we have multiple choices where multiple outputs are generated for each decoding time-step. We have decided to use the concatenation of the embeddings of both generated symbols based on the work [9].

The FNMT model may lead to sequences with a different length, since lemmas and factors are generated simultaneously but separately (each sequence ends after the generation of the end of sequence <*eos*> token). To avoid this, the sequences length is decided based on the lemmas stream length (i.e. the length of the factors sequence is constrained to be equal to the length of the lemma sequence). This is motivated by the fact that the lemmas contain most of the information of the final surface form (word).

Once we obtain the factored outputs from the neural network, we need to combine them to obtain the surface form (word representation). This operation is also performed with the MACAON tool, which, given a lemma and some factors, provides the word. Word forms given by MACAON toolkit have a 99% success rate. In the cases (e.g. name entities) that the word corresponding to the lemma and factors is not found, the system outputs the lemma itself.

4 Experiments

We performed a set of experiments for Factored NMT (FNMT) and compared them with the word-based NMT and BPE-based NMT systems.

4.1 Data Processing and Selection

The systems are trained on the English to French (EN-FR) Spoken Language Translation task from IWSLT 2015 evaluation campaign[1]. We applied data selection using modified Moore-Lewis filtered by XenC [21] to obtain a sub part of the available parallel corpora (news-commentary, united-nations, europarl, wikipedia, and two crawled corpora). The Technology Entertainment Design (TED) [4] corpus has been used as in-domain corpus.

We preprocess the data to convert html entities and filter out the sentences with more than 50 words for both source and target languages. Finally, we obtain a corpus of 2M sentences with 147k unique words for the English side and 266k unique words for the French side. French vocabulary is bigger than English since French is more highly inflected language. Table 1 shows training, development and testing sets statistics.

Table 1. Datasets statistics

Data	Corpus name	Datasets	# Sents	# Words EN-FR
Training	train15	data selection	2M	147–266k
Development	dev15	dev10 + test10 + test13	3.6k	7.3–8.9k
Testing	test15	test11 + test12	1.9k	4.5–5.4k

4.2 Training

Models are trained using NMTPY [3], an NMT toolkit in Python based on Theano[2]. The following hyperparameters have been chosen to train the systems. The embedding and recurrent layers have the dimensions 620 and 1000, respectively. The batch size is set to 80 sentences and the parameters are trained using the Adadelta [27] optimizer. We clipped the norm of the gradient to be no more than 1 [20] and initialize the weights using *Xavier* [10]. The systems are validated on dev15 dataset using early stopping based on BLEU [19]. The vocabulary size of the source language is set to 30K. The output layer size of the baseline NMT system is set to 30K. For the sake of comparability and consistency, the same value (30k) is used for the lemma output of the FNMT system. This 30K FNMT vocabulary includes 17k lemmas obtained from the original NMT vocabulary (30k word level gives 17k lemmas when all the derived forms

[1] https://sites.google.com/site/iwsltevaluation2015.
[2] https://github.com/lium-lst/nmtpy.

of the verbs, nouns, adjectives, etc. are discarded) increased with additional new lemmas to fit the 30K desired value. The factors have 142 different units in their vocabulary. When it comes to combining the lemmas and the factors vocabulary, the system is able to generate 172K different words, using the external linguistic resources, which is 6 times bigger than a standard word-based NMT vocabulary.

For BPE systems, bilingual vocabulary has been built using source and target language applying the joint vocabulary BPE approach. In order to create comparable BPE systems, we set the number of merge operations for the BPE algorithm (the only hyperparameter of the method) as 30K minus the number of character according to the paper [24]. Then, we apply a total of 29388 merge operations to learn the BPE models on the training and validation sets. During the decoding process, we use a beam size of 12 as used in [2].

4.3 Quantitative Results

The Factored NMT system aims at integrating linguistic knowledge into the decoder in order to overcome the restriction of having a large vocabulary at target side. We first compare our system with the standard word-level NMT system. For the sake of comparison with state of the art systems, we have built a subword system using the BPE method. Subwords were calculated at the input and the output side of the neural network as described in [24]. The results are measured with two automatic metrics, the most common metric for machine translation BLEU and METEOR [13]. We evaluate on test15 dataset from the IWSLT 2015 campaign and results are presented in Table 2.

Table 2. Results on IWSLT test15. %BLEU and %METEOR performance of NMT and FNMT systems with and without UNK replacement (UR) are presented. For each system we provide the number of generated UNK tokens in the last column

Model	%METEOR↑	%BLEU↑			#UNK
	Word	Word	Lemma	Factors	
NMT/+UR	62.21/63.38	41.80/42.74	45.10	51.80	1111
BPE	62.87	42.37	45.96	53.31	0
FNMT/+UR	**64.10/64.81**	**43.42/44.15**	**47.18**	**54.24**	**604**

As we can see from the Table 2 results, the FNMT system obtains better %BLEU and %METEOR scores compared to the state of the art NMT and BPE systems. An improvement of about **1** %BLEU point is achieved compared to the best baseline system (BPE). This improvement is even bigger (**1.4** %BLEU point) when UNK replacement is applied to both systems. In a quest to better understand the reasons of this improvement, we also computed the %BLEU scores of each output level (lemmas and factors) for FNMT. Theses scores are presented in Table 2. The lemma and factors scores of NMT and BPE systems

are obtained through a decomposing of their word level output into lemma and factors. We observe yet again that FNMT systems gives better score at both lemma and factors level. Replacement of unknown words has been performed using the alignments extracted from the attention mechanism. We have replaced the generated UNK tokens by translating its highest probability aligned source word. We see an improvement of around 1 point %BLEU score in both NMT and FNMT systems.

The last column of Table 2 shows, for each system, the number of generated UNK tokens. As shown in the table our FNMT system produces half of the UNK tokens compared to the word-based NMT system. This tends to prove that the Factored NMT system effectively succeed in modelling more words compared to the word based NMT system augmenting the generalization power of our model and preserving manageable output layer sizes. Though we can see that BPE system does not produce UNK tokens, this is not reflected in the scores. Indeed, this can be due to the possibility of generation of incorrect words using BPE units in contrast to the FNMT system.

4.4 Qualitative Analysis

The strengths of FNMT are considered under this qualitative analysis. We have studied and compared the translation outputs of NMT at word-level and BPE-level with the ones of FNMT systems. Two examples are presented in Fig. 3 and Table 3.

Table 3. Examples of translations with NMT, BPE and FNMT systems (without unknown words replacement)

Src	we in medicine , I think , are baffled									
Ref	Je pense que en médecine nous sommes dépassés									
NMT	Nous	,	en	médecine	,	je	pense	,	sont	UNK
BPE	nous	,	en	médecine	,	je	pense	,	sommes	b@@ af@@ és
FNMT	nous	,	en	médecine	,	je	pense	,	sont	déconcertés
Lemmas	lui	,	en	médecine	,	je	penser	,	être	déconcerter
Factors	pro-1-p-l	pct-l	prep-l	nc-f-s-l	pct-l	cln-1-s-l	v-P-1-s-l	pct-l	v-P-3-p-l	vppart-K-m-p-l

The reference translation of the source sentence presented in Fig. 3 is *"mais voilà où ça devient intéressant"*. As we can see, contrary to the baseline NMT system, the FNMT system matches exactly the reference and thus produces the correct translation. An additional interesting observation is that the alignment provided by the attention mechanism seems to be better defined and more helpful when using factors. Also, one can notice the difference between the attention distributions made by the systems over the source sentence. The NMT system first translated "here" into "là", added a coma, and then was in trouble for translating the rest of the sentence, which is reflected by the rather fuzzy attention weights. The FNMT system had better attention distribution over of the source sentence in this case.

Table 3 shows another example comparing NMT, BPE and FNMT systems. The NMT system generated an unknown token (UNK) when translating the English word *"baffled"*. We observe that BPE translates *"baffled"* to *"bafs"* which does not exist in French. This error probably comes from the shared vocabulary between the source and target languages creating an incorrect word very similar to its aligned source tokens. FNMT translates it to *"dconcerts"* which is a better translation than in the reference. One should note that it is not generated by the unknown word replacement method. However, for this particular example, an error on the factors leads to the word *"sont"* instead of *"sommes"*, resulting in lower automatic scores for FNMT output.

5 Conclusion

In this paper, the Factored NMT approach has been further explored. Factors based on linguistic *a priori* knowledge have been used to decompose the target words. This approach outperforms a strong baseline system using subword units computed with byte pair encoding. Our FNMT system is able to model an almost 6 times bigger word vocabulary with only a slight increase of the computational cost. By these means, the FNMT system is able to halve the generation of unknown tokens compared to word-level NMT. Using a simple unknown word replacement procedure involving a bilingual dictionary, we are able to obtain even better results (+0.8 %BLEU compared to previous best system).

Also, the use of external linguistic resources allows us to generate new word forms that would not be included in the standard NMT system *shortlist*. The advantage of this approach is that the new generated words are controlled by the linguistic knowledge, that avoid producing incorrect words, as opposed to actual systems using BPE. We demonstrated the performance of such a system on an inflected language (French). The results are very promising for use with highly inflected languages like Arabic or Czech.

Acknowledgments. This work was partially funded by the French National Research Agency (ANR) through the CHIST-ERA M2CR project, under the contract number ANR-15-CHR2-0006-01.

References

1. Alexandrescu, A.: Factored neural language models. In: HLT-NAACL (2006)
2. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. CoRR abs/1409.0473 (2014)
3. Caglayan, O., García-Martínez, M., Bardet, A., Aransa, W., Bougares, F., Barrault, L.: NMTPY: a flexible toolkit for advanced neural machine translation systems. arXiv preprint arXiv:1706.00457 (2017)
4. Cettolo, M., Girardi, C., Federico, M.: WIT^3: web inventory of transcribed and translated talks. In: Proceedings of the 16th Conference of the European Association for Machine Translation (EAMT), Trento, Italy, pp. 261–268, May 2012

5. Cho, K., van Merrienboer, B., Gülçehre, Ç., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using RNN encoder-decoder for statistical machine translation. CoRR abs/1406.1078 (2014)
6. Chung, J., Cho, K., Bengio, Y.: A character-level decoder without explicit segmentation for neural machine translation. CoRR abs/1603.06147 (2016)
7. Costa-Jussà, M.R., Fonollosa, J.A.R.: Character-based neural machine translation. CoRR abs/1603.00810 (2016)
8. Firat, O., Cho, K.: Conditional gated recurrent unit with attention mechanism (2016). github.com/nyu-dl/dl4mt-tutorial/blob/master/docs/cgru.pdf
9. García-Martínez, M., Barrault, L., Bougares, F.: Factored neural machine translation architectures. In: Proceedings of the International Workshop on Spoken Language Translation, IWSLT 2016, Seattle, USA (2016). http://workshop.2016.iwslt.org/downloads/IWSLT_2016_paper_2.pdf
10. Glorot, X., Bengio, Y.: Understanding the difficulty of training deep feedforward neural networks. In: Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS 2010). Society for Artificial Intelligence and Statistics (2010)
11. Jean, S., Cho, K., Memisevic, R., Bengio, Y.: On using very large target vocabulary for neural machine translation. CoRR abs/1412.2007 (2014)
12. Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., Herbst, E.: Moses: open source toolkit for statistical machine translation. In: Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions, ACL 2007, pp. 177–180. Association for Computational Linguistics, Stroudsburg (2007)
13. Lavie, A., Agarwal, A.: METEOR: an automatic metric for mt evaluation with high levels of correlation with human judgments. In: Proceedings of the Second Workshop on Statistical Machine Translation, StatMT 2007, pp. 228–231. Association for Computational Linguistics, Stroudsburg (2007)
14. Le, H.S., Oparin, I., Messaoudi, A., Allauzen, A., Gauvain, J.L., Yvon, F.: Large vocabulary SOUL neural network language models. In: INTERSPEECH (2011). sources/Le11large.pdf
15. Ling, W., Trancoso, I., Dyer, C., Black, A.W.: Character-based neural machine translation. CoRR abs/1511.04586 (2015)
16. Luong, T., Sutskever, I., Le, Q.V., Vinyals, O., Zaremba, W.: Addressing the rare word problem in neural machine translation. CoRR abs/1410.8206 (2014). http://arxiv.org/abs/1410.8206
17. Nasr, A., Béchet, F., Rey, J.F., Favre, B., Roux, J.L.: MACAON, an NLP tool suite for processing word lattices. In: Proceedings of the ACL-HLT 2011 System Demonstrations, pp. 86–91 (2011)
18. Niehues, J., Ha, T.L., Cho, E., Waibel, A.: Using factored word representation in neural network language models. In: Proceedings of the First Conference on Machine Translation, pp. 74–82. Association for Computational Linguistics, Berlin, August 2016
19. Papineni, K., Roukos, S., Ward, T., Zhu, W.J.: BLEU: a method for automatic evaluation of machine translation. In: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, ACL 2002, Stroudsburg, PA, USA, pp. 311–318 (2002)
20. Pascanu, R., Mikolov, T., Bengio, Y.: Understanding the exploding gradient problem. CoRR abs/1211.5063 (2012)

21. Rousseau, A.: XenC: an open-source tool for data selection in natural language processing. Prague Bull. Math. Linguist. **100**, 73–82 (2013)
22. Sennrich, R.: How grammatical is character-level neural machine translation? Assessing MT quality with contrastive translation pairs. CoRR abs/1612.04629 (2016)
23. Sennrich, R., Haddow, B.: Linguistic input features improve neural machine translation. CoRR abs/1606.02892 (2016)
24. Sennrich, R., Haddow, B., Birch, A.: Neural machine translation of rare words with subword units. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics. Long Papers, vol. 1, pp. 1715–1725. Association for Computational Linguistics (2016)
25. Sutskever, I., Vinyals, O., Le, Q.V.: Sequence to sequence learning with neural networks. CoRR abs/1409.3215 (2014)
26. Wu, Y., Yamamoto, H., Lu, X., Matsuda, S., Hori, C., Kashioka, H.: Factored recurrent neural network language model in TED lecture transcription. In: IWSLT (2012)
27. Zeiler, M.D.: ADADELTA: an adaptive learning rate method. arXiv preprint arXiv:1212.5701 (2012)