One Size Does Not Fit All: Comparing NMT Representations of Different Granularities

Nadir Durrani **Fahim Dalvi** **Hassan Sajjad** **Yonatan Belinkov*** **Preslav Nakov**
{ndurrani,faimaduddin,hsajjad,pnakov}@qf.org.qa
Qatar Computing Research Institute, HBKU Research Complex, Doha 5825, Qatar

*MIT Computer Science and Artificial Intelligence Laboratory and
Harvard John A. Paulson School of Engineering and Applied Sciences, Cambridge, MA, USA
belinkov@mit.edu

Abstract

Recent work has shown that contextualized word representations derived from neural machine translation are a viable alternative to such from simple word predictions tasks. This is because the internal understanding that needs to be built in order to be able to translate from one language to another is much more comprehensive. Unfortunately, computational and memory limitations as of present prevent NMT models from using large word vocabularies, and thus alternatives such as subword units (BPE and morphological segmentations) and characters have been used. Here we study the impact of using different kinds of units on the quality of the resulting representations when used to model morphology, syntax, and semantics. We found that while representations derived from subwords are slightly better for modeling syntax, character-based representations are superior for modeling morphology and are also more robust to noisy input.

1 Introduction

Recent years have seen the revolution of deep neural networks and the subsequent rise of representation learning based on network-internal activations. Such representations have been shown useful when addressing various problems from fields such as image recognition (He et al., 2016), speech recognition (Bahdanau et al., 2016), and natural language processing (NLP) (Mikolov et al., 2013a). The central idea is that the internal representations trained to solve an NLP task could be useful for other tasks as well. For example, word embeddings learned for a simple word prediction task in context, word2vec-style (Mikolov et al., 2013b), have now become almost obligatory in state-of-the-art NLP models. One issue with such word embeddings is that the resulting representation is context-independent.

Recently, it has been shown that huge performance gains can be achieved by contextualizing the representations, so that the same word could have a different embedding in different contexts. This is best achieved by changing the auxiliary task. For example, ELMo (Peters et al., 2018) learns contextualized word embeddings from language modeling (LM) using long short-term memory networks (LSTMs) (Hochreiter and Schmidhuber, 1997).

It has been further argued that complex auxiliary tasks such as neural machine translation (NMT) are better tailored for representation learning, as the internal understanding of the input language that needs to be built by the network to be able to translate from one language to another needs to be much more comprehensive compared to what would be needed for a simple word prediction task. This idea is implemented in the seq2seq-based CoVe model (McCann et al., 2017).

More recently, the BERT model (Devlin et al., 2019) proposed to use representation from another NMT model, the Transformer, while optimizing for two LM-related auxiliary tasks: (*i*) masked language model and (*ii*) next sentence prediction.

Another important aspect of representation learning is the basic unit the model operates on. In word2vec-style embeddings, it is the word, but this does not hold for NMT-based models, as computational and memory limitations, as of present, prevent NMT from using a large vocabulary, typically limiting it to 30-50k words (Wu et al., 2016). This is a severe limitation, as most NLP applications need to handle vocabularies of millions of words, e.g., word2vec (Mikolov et al., 2013b), GloVe (Pennington et al., 2014) and FastText (Mikolov et al., 2018) offer pre-trained embeddings for 3M, 2M, and 2.5M words/phrases. The problem is typically addressed using byte-pair encoding (BPE), where words are segmented into pseudo-word sequences (Sennrich et al., 2016).

Proceedings of NAACL-HLT 2019, pages 1504–1516
Minneapolis, Minnesota, June 2 - June 7, 2019.

A less popular solution is to use characters as the basic unit (Chung et al., 2016; Lee et al., 2017), and in the case of morphologically complex languages, yet another alternative is to reduce the vocabulary size by using unsupervised morpheme segmentation (Bradbury and Socher, 2016).

The impact of using different units of representation in NMT models has been studied in previous work (Ling et al., 2015; Costa-jussà and Fonollosa, 2016; Chung et al., 2016; Lee et al., 2017, among others), but the focus has been exclusively on the quality of the resulting translation output. However, it remains unclear what input and output units should be chosen if we are primarily interested in representation learning. Here, we aim at bridging this gap by evaluating the quality of NMT-derived embeddings originating from units of different granularity when used for modeling morphology, syntax, and semantics (as opposed to end tasks such as sentiment analysis and question answering). Our contributions are as follows:

- We study the impact of using words vs. characters vs. BPE units vs. morphological segments on the quality of representations learned by NMT models when used to model morphology, syntax, and semantics.
- We further study the robustness of these representations with respect to noise.

We found that while representations derived from morphological segments are better for modeling non-local syntactic and semantic dependencies, character-based ones are superior for morphology and are also more robust to noise. There is also value in combining different representations.

2 Related Work

Representation analysis aims at demystifying what is learned inside the neural network blackbox. This includes analyzing word and sentence embeddings (Adi et al., 2017; Qian et al., 2016b; Ganesh et al., 2017; Conneau et al., 2018, among others), RNN states (Qian et al., 2016a; Shi et al., 2016; Wu and King, 2016; Wang et al., 2017), and NMT representations (Shi et al., 2016; Belinkov et al., 2017a), as applied to morphological (Vylomova et al., 2017; Dalvi et al., 2017), semantic (Qian et al., 2016b; Belinkov et al., 2017b) and syntactic (Linzen et al., 2016; Tran et al., 2018; Conneau et al., 2018) tasks. See Belinkov and Glass (2019) for a recent survey.

Other studies carried a more fine-grained neuron-level analysis for NMT and LM (Dalvi et al., 2019; Bau et al., 2019; Lakretz et al., 2019). While previous work focused on words, here we compare units of different granularities.

Subword translation units aim at reducing the vocabulary size and the out-of-vocabulary (OOV) rate. Researchers have used BPE units (Sennrich et al., 2016), morphological segmentation (Bradbury and Socher, 2016), characters (Durrani et al., 2014; Lee et al., 2017), and hybrid units (Ling et al., 2015; Costa-jussà and Fonollosa, 2016) to address the OOV word problem in MT. The choice of translation unit impacts what the network learns. Sennrich (2017) carried a systematic error analysis by comparing subword versus character units and found the latter to be better at handling OOV and transliterations, whereas BPE-based subword units were better at capturing syntactic dependencies. In contrast, here we focus on representation learning, not translation quality.

Robustness to noise is an important aspect in machine learning. It has been studied for various models (Szegedy et al., 2014; Goodfellow et al., 2015), including NLP in general (Papernot et al., 2016; Samanta and Mehta, 2017; Liang et al., 2018; Jia and Liang, 2017; Ebrahimi et al., 2018; Gao et al., 2018), and character-based NMT in particular (Heigold et al., 2018; Belinkov and Bisk, 2018). Unlike this work, we compare robustness to noise for units of different granularity. Moreover, we focus on representation learning rather than on the quality of the translation output.

3 Methodology

Our methodology is inspired by research on interpreting neural network (NN) models. A typical framework involves extracting feature representations from different components (e.g., encoder/decoder) of a trained model and then training a classifier to make predictions for an auxiliary task. The performance of the trained classifier is considered to be a proxy for judging the quality of the extracted representations with respect to the particular auxiliary task.

Formally, for each input word $\mathbf{x}_i$ we extract the corresponding LSTM hidden state(s) from each layer of the encoder/decoder. We then concatenate the representations of the layers and use them as a feature vector $\mathbf{z}_i$ for the auxiliary task.

Words	Obama	receives	Netanyahu	in	the	capital	of	USA
POS	NP	VBZ	NP	IN	DT	NN	IN	NP
Sem.	PER	ENS	PER	REL	DEF	REL	REL	GEO
CCG	NP	((S[dcl]\NP)/PP)/NP	NP	PP/NP	PP/N	N	(NP\NP)/NP	NP

Table 1: Example sentence with different annotations.

Words	He admits to shooting girlfriend
BPE	He admits to sho@@ oting gir@@ l@@ friend
Morfessor	He admit@@ s to shoot@@ ing girl@@ friend
Characters	H e _ a d m i t s _ t o _ s h o o t i n g _ g i r l f r i e n d

Table 2: Example with different segmentations.

We then train a logistic regression classifier, minimizing the cross-entropy loss:

$$\mathcal{L}(\theta) = -\sum_i \log P_\theta(\mathbf{l}_i|\mathbf{x}_i)$$

where $P_\theta(\mathbf{l}|\mathbf{x}_i) = \frac{\exp(\theta_l \cdot \mathbf{z}_i)}{\sum_{l'} \exp(\theta_{l'} \cdot \mathbf{z}_i)}$ is the probability that word $\mathbf{x}_i$ is assigned label $\mathbf{l}$.

We learn the weights $\theta \in \mathbb{R}^{D \times L}$ using gradient descent. Here D is the dimensionality of the latent representations $\mathbf{z}_i$ and L is the size of the label set for property $\mathcal{P}$. See Section 4 for details.

3.1 Word Representation Units

We consider four representation units: words, byte-pair encoding (BPE) units, morphological units, and characters. Table 2 shows an example of each representation unit. *BPE* splits words into symbols (a symbol is a sequence of characters) and then iteratively replaces the most frequent sequences of symbols with a new merged symbol. In essence, frequent character n-grams merge to form one symbol. The number of merge operations is controlled by a hyper-parameter `OP`; a high value of `OP` means coarse segmentation and a low value means fine-grained segmentation (Sajjad et al., 2017). For *morphologically segmented units*, we use an unsupervised morphological segmenter, Morfessor (Smit et al., 2014). Note that although BPE and Morfessor segment words at a similar level of granularity, the segmentation generated by Morfessor is linguistically motivated. For example, it splits the gerund verb *shooting* into root *shoot* and the suffix *ing*. Compare this to the BPE segmentation *sho* + *oting*, which has no linguistic connotation. On the extreme, the fully *character-level* units treat each word as a sequence of characters.

3.2 Extracting Activations for Subword and Character Units

Previous work on analyzing NMT representations has been limited to the analysis of word representations only,[1] where there is a one-to-one mapping from input units (words) and their NMT representations (hidden states) to their linguistic annotations (e.g., morphological tags).

In the case of subword-based systems, each word may be split into multiple subword units, and each unit has its own representation. It is less trivial to define which representations should be evaluated when predicting a word-level linguistic property such as part of speech. We consider two simple approximations to estimate a word representation from subword units:

(*i*) **Average**: for each source (or target) word, we average the activation values of all the subwords (or characters) comprising it. In the case of a bi-directional encoder, we concatenate the averages from the forward and the backward activations of the encoder on the subwords (or characters) that represent the current word.[2]

(*ii*) **Last**: we consider the activation of the last subword (or character) as the representation of the word. For the bi-directional encoder, we concatenate the forward encoder's activation on the last subword unit with the backward encoder's activation on the first subword unit.

This formalization allows us to analyze the quality of character- and subword-based representations at the word level via prediction tasks. Such kind of analysis has not been performed before.

[1] Belinkov et al. (2017a) analyzed representations trained from character CNN models (Kim et al., 2016), but the extracted features were still based on word representations produced by the character CNN. As a result, they could not analyze and compare results for the BPE and character-based models that do not assume segmentation into words.

[2] One could envision more sophisticated averages, such as weighting via an attention mechanism.

4 Linguistic Properties

We choose three fundamental NLP tasks that serve as a good representative of various properties inherent in a language, ranging from morphology (word structure), syntax (grammar), and semantics (meaning). In particular, we experiment with *morphological tagging* for German, Czech, Russian, and English,[3] *lexical semantic tagging* for English and German, and *syntactic tagging* via CCG supertagging for English. Table 1 shows an example sentence with annotations for each task.

The morphological tags capture word structure, the semantic tags reflect lexical semantics, and the syntactic tags (CCG supertags) capture global syntactic information locally, at the lexical level.

For example, in Table 1, the morphological tag `VBZ` for the word *receives* marks it as a verb in third person, singular, present tense; the semantic tag `ENS` describes a *present simple event* category; and the syntactic tag `PP/NP`) can be thought of as a function that takes a noun phrase on the right (e.g., *the capital of USA*), and returns a prepositional phrase (e.g., *in the capital of USA*).

Artificial Error Induction Recent studies have shown that small perturbations in the input can cause significant deterioration in the performance of the deep neural networks. Here, we evaluate the robustness of various representations under noisy input conditions. We use corpora of real errors harvested by Belinkov and Bisk (2018). The errors contain a good mix of typos, misspellings, and other kinds of errors. In addition, we created data with synthetic noise. We induced two kinds of errors: (*i*) *swap* and (*ii*) *middle*. *Swap* is a common error, which occurs when neighboring characters are mistakenly swapped, e.g., *word* → *wodr*. In *Middle* errors, the order of the first and the last characters of a word are preserved, while the middle characters are randomly shuffled (Rawlinson, 1976), e.g., *example* → *eaxmlpe*. We corrupt $n\%$ words randomly in each test sentence, using *swap* or *middle* heuristics, or replace words using real-error corpora. We then re-extract feature vectors for the erroneous words in a sentence and we re-evaluate the prediction capability of these embeddings on the linguistic tasks.

[3] As English is morphologically poor, we use part-of-speech tags for it. We refer to English part-of-speech tags as morphological tags later in the paper in order to keep the terminology consistent.

	de-en	cs-en	ru-en
Train	507K	340K	370K
Dev	3,000	3,000	2,818
Test	3,000	3,000	2,818

(a) NMT data

	de	cs	ru	en
Morphology	509	1,004	602	42
Semantics	69	–	–	66
Syntax	1,272	–	–	–

(b) Number of tags

	de	cs	ru	en
	Morphology			
Train	14,498	14,498	11,824	14498
Test	8,172	8,172	6,000	8,172
	Semantics			
Train	–	–	–	14,084
Test	–	–	–	12,168
CV	1,863	–	–	–
	Syntax			
Train	–	–	–	41,586
Test	–	–	–	2,407

(c) Classifier data

Table 3: Statistics about NMT and classifier training data for English (en), German (de), Russian (ru), and Czech (cs). Here, CV stands for cross-validation.

5 Experimental Setup

Data and Languages We trained NMT systems for four language pairs: German-English, Czech-English, Russian-English, and English-German, using data made available through two popular machine translation campaigns, namely, WMT (Bojar et al., 2017) and IWSLT (Cettolo et al., 2016). We trained the MT models using a concatenation of the NEWS and the TED training datasets, and we tested on official TED test sets (testsets-11-13) to perform the evaluation using BLEU (Papineni et al., 2002). We trained the morphological classifiers and we tested them on a concatenation of the NEWS and the TED testsets, which were automatically tagged as described in the next paragraph. We trained and evaluated the semantic and the syntactic classifiers on existing annotated corpora. See Table 3 for details about the datasets.

Taggers We used RDRPOST (Nguyen et al., 2014) to annotate data for the classifier. For semantic tagging, we used the gold-annotated semantic tags from the Groningen Parallel Meaning Bank (Abzianidze et al., 2017), which were made available by (Bjerva et al., 2016). The tags are grouped into coarse categories such as events, names, time, and logical expressions. There is enough data for English (≈42K), and we randomly sampled the same amount of data we used to train our morphological classifiers to train the semantic classifiers. Yet, only 1,863 annotated sentences (12,783 tokens) were available for German. Thus, in the experiments, we performed 5-fold cross-validation. For CCG supertagging, we used the English CCGBank (Hockenmaier and Steedman, 2007), which contains 41,586/2,407 train/test sentences.[4] See Table 3 for more detailed statistics about the train/dev/test datasets we used.

MT Systems and Classifiers We used `seq2seq-attn` (Kim, 2016) to train a two-layer encoder-decoder NMT model based on LSTM representation with attention (Hochreiter and Schmidhuber, 1997) with a bidirectional encoder and a unidirectional decoder.[5] We used 500 dimensions for both word embeddings and LSTM states. We trained the systems with SGD for 20 epochs and we used the final model, i.e., the one with the lowest loss on the development dataset, to generate features for the classifier.

We trained our neural machine translation models in both *-to-English and English-to-* translation directions, and we analyzed the representations from both the *encoder* and the *decoder*. In order to analyze the representations derived from the encoder side, we fixed the decoder side with BPE-based embeddings, and we trained the source side with word/BPE/Morfessor/character units. Similarly, when analyzing the representations from the decoder side, we trained the encoder representation with BPE units, and we varied the decoder side using word/BPE/char units. Our motivation for this setup is that we wanted to analyze the encoder/decoder side representations in isolation, keeping the other half of the network (i.e., the decoder/encoder) static across different settings.[6]

[4]There are no available CCG banks for the other languages we experiment with, except for a German CCG bank, which is not publicly available (Hockenmaier, 2006).

[5]The decoder has to be unidirectional as, at decoding time, the future is unknown.

[6]Heigold et al. (2018) used a similar setup.

In our experiments, we used 50k BPE operations and we limited the vocabulary of all systems to 50k. Moreover, we trained the word, BPE, Morfessor, and character-based systems with maximum sentence lengths of 80, 100, 100, and 400 units, respectively.

For the classification tasks, we used a logistic regression classifier whose input is either the hidden states in the case of the word-based models, or the **Last** or the **Average** representations in the case of character- and subword-based models. Since for the bidirectional encoder we concatenate forward and backward states from all layers, this yields 2,000/1,000 dimensions when classifying using the representations from the encoder/decoder: 500 dimensions×2 layers×2 directions (1 for the decoder, as it is uni-directional). In all cases, we trained the logistic regression classifier for ten epochs.

6 Results

We now present the evaluation results for using representations learned from different input units to predict morphology, semantics, and syntax. For subword and character units, we found the activation of the last subword/character unit of a word to be consistently better than using the average of all activations (See Table 4). Therefore, we report only the results using the **Last** method, for the remainder of the paper.

	de		cs		ru	
	sub	char	sub	char	sub	char
Last	78.5	80.5	78.6	88.3	80.0	88.8
Avg	76.3	79.2	76.4	84.9	78.3	84.4

Table 4: Classifier accuracy for the representations generated by aggregating subword (sub) or character (char) representations using either the average or the last LSTM state for each word.

6.1 Morphological Tagging

Figure 1 summarizes the results for predicting morphological tags with representations learned using different units. The character-based representations consistently outperformed other representations on all language pairs, while the word-based ones performed worst. The differences are more significant in the case of languages with relatively complex morphology such as Czech.

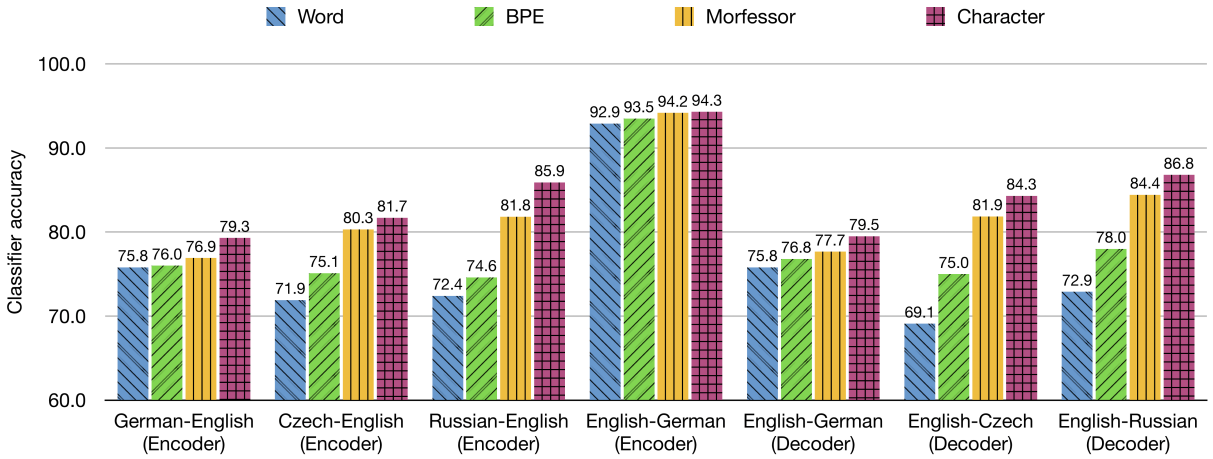


Figure 1: Morphological tagging accuracy across various systems and language pairs. The first four groups of results use BPE on the decoder side, while the last three groups use BPE on the encoder side.

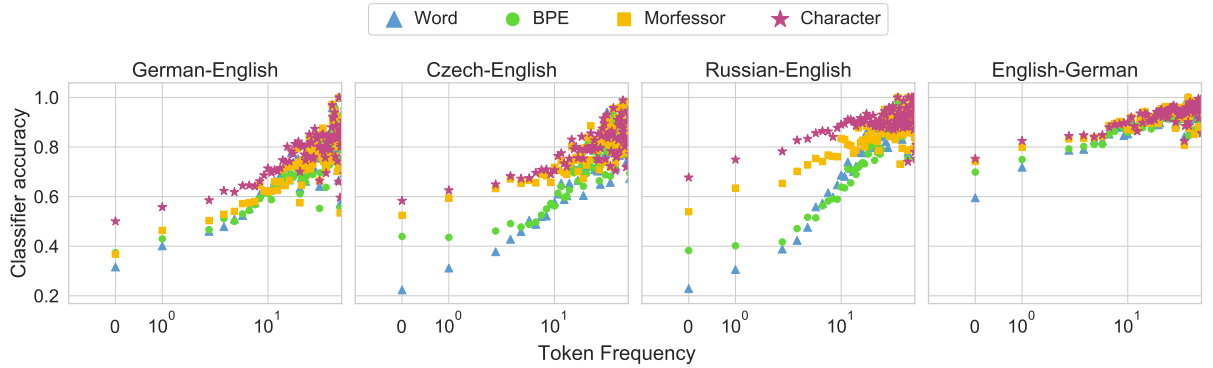


Figure 2: Morphological tagging accuracy vs. word frequency for different encoding units. Best viewed in color.

	de-en	cs-en	ru-en	en-de
word → bpe	34.0	27.5	20.9	29.7
bpe → bpe	35.6	28.4	22.4	30.2
morf → bpe	35.5	28.5	22.5	29.9
char → bpe	34.9	29.0	21.3	30.0

Table 5: BLEU scores across language pairs. "**s** → t" means **s**ource and **t**arget units; morf = Morfessor.

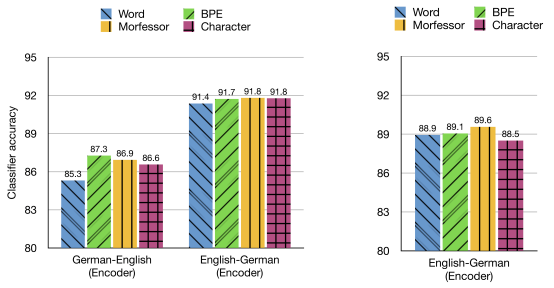
	de-en	cs-en	ru-en	en-de
MT	3.42	6.46	6.86	0.82
Classifier	4.42	6.13	6.61	2.09

Table 6: OOV rate for the MT and the classifier testsets.

We see in Figure 1 a difference of up to 14% in favor of character-based representations when compared with word-based ones. The improvement is minimal in the case of English (1.2%), which is a morphologically simpler language. This is also somewhat reflected in the translation quality.

We can see in Table 5 that character-based segmentation yielded higher BLEU scores in the case of a morphologically rich language such as Czech, but performed poorly in the case of German, which requires handling long-distance dependencies. Comparing subword units, we found Morfessor to yield much better morphological tagging performance, especially in the case of morphologically rich languages such as Czech and Russian, supposedly due to the Morfessor's linguistically motivated segmentations, which are helpful for learning morphology.

We further investigated whether the performance difference between the representations is due to the difference in modeling infrequent and out-of-vocabulary (OOV) words. Table 6 shows the OOV rate for each language, which is higher for morphologically rich languages. Figure 2 shows that the gap between different representations is inversely proportional to the frequency of



(a) Semantic tagging. (b) Syntactic tagging.

Figure 3: Semantic and syntactic tagging for English and German (using BPE on the decoder).

the word in the training data, and character-based models handle infrequent and OOV words better.

Decoder Representations Next, we used the decoder representations from the English-to-* models. We saw similar performance trends as for the encoder-side representations: characters performed best, and words performed worst. Again, the morphological units performed better than the BPE-based units. Comparing encoder representations to decoder representations, it is interesting to see that in several cases the decoder side representations performed better than the encoder side ones, even though the former were trained using a uni-directional LSTM. However, since there is no difference in the general trends between the encoder- and the decoder-side representations, below we focus on the encoder-side only.

6.2 Semantic Tagging

Figure 3a summarizes the experimental results for evaluating representtaion units of the semantic tagging task. For English, the subword (BPE and Morfessor) and the character representations yielded comparable results. However, for German, BPE performed better. This is in contrast with the morphology prediction experiments, where the character representations were consistently better. We will discuss this in more detail in Section 7.

6.3 Syntactic Tagging

The final task we experimented with is CCG super-tagging, which reflects modeling syntactic structure. Here we only have English tags, and thus we evaluate the performance of encoder representations for English→German models, trained using words, characters, and subword units.

We can see in Figure 3b that the morphologically segmented representation units performed the best overall. Moreover, there is no much difference when using word-based vs. BPE-based representations.

The character-based representations lag behind, but the difference in accuracy is small compared to the morphological tagging results.[7] It is noteworthy that here character-based representations perform worse than both words and subwords, contrary to their superior performance on morphology. We will return to this in Section 7 below.

6.4 Robustness to Noise

Next, we evaluated the robustness of the representations with respect to noise. We induced errors in the test sets by corrupting 25% of the words in each sentence using different error types (synthetic or real noise), as described in Section 4. We extracted the representations of the noisy test sets and we re-evaluated the classifiers. Figure 4 shows the performance on each task. We can see that characters yielded much better performance on all tasks and for all languages, showing minimal drop in accuracy, in contrast to earlier results where they did not outperform subword units[8] on the task of syntactic tagging. This shows that character representations are more robust to noise.

Surprisingly, in a few cases, BPE performed worse than word units, e.g., in the case of syntactic tagging (80.3 vs. 81.1). We found that BPE can segment a noisy word into two or more known subword units that have no real relationship to the actual word. Thus, using representations of wrong subword units could hurt the performance.

We further investigated the robustness of each classifier by increasing the percentage of noise in the test data. We found that the difference in representation quality stays constant across BPE and character representations, whereas word representations deteriorate significantly as the amount of noise increases (see Figure 5).

7 Discussion

7.1 Performance Across Various Tasks

Our experiments show a complicated picture, where none of the representations is superior in all scenarios. Characters were found to be better for morphological tagging, BPE was ahead in

[7]For perspective, these numbers are above a majority class baseline of 72% and below the state-of-the-art, which is around 94-95% (Kadari et al., 2018; Xu, 2016).

[8]Morphological segmentation showed similar results compared to BPE-based segmentation in these experiments.

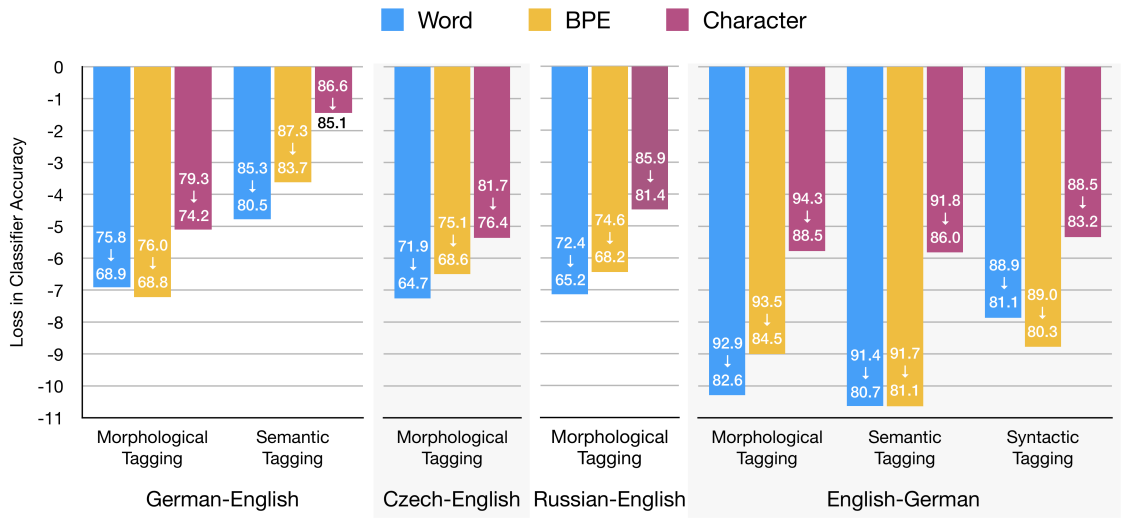
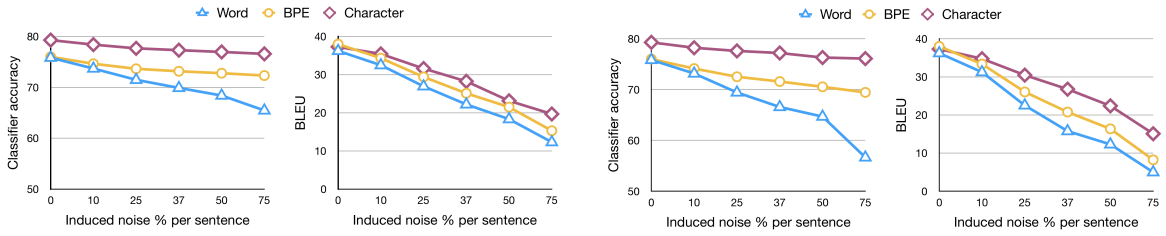


Figure 4: Classification accuracy for morphological, syntactic and semantic tagging with 25% noise in each sentence. Absolute scores (original → noisy) are shown inset.



(a) Real noise.

(b) Synthetic noise.

Figure 5: Results for morphological classification when adding induced noise.

the semantic tagging task for German (and about the same in English), and Morfessor units were slightly better for syntax.

Syntactic tagging requires knowledge about the complete sentence. Splitting a sentence into characters substantially increases its length: on average from 50 words to 250 single-character tokens. Thus, character-based models struggle to capture long-distance dependencies. Sennrich (2017) also found this to be true in their evaluation based on contrastive translation pairs in German-English.

Similarly, in the case of morphological tagging, the information about the morphological structure of a word is dependent on the surrounding words plus some internal information (root, morphemes, etc.) present inside the word. A character-based system has access to all of this information, and thus performs well. Morphological segmentation performed better than BPE for the morphological tagging because its segments are linguistically motivated units (segmented into root + morphemes), thus making the information about the word morphology explicit in the representation. In contrast, BPE solely focuses on the frequency of characters occurring together in the corpus, and thus can generate linguistically incorrect units.

		W	B	C	W+B	B+C	W+C	ALL
Morph	DE	75.8	76.0	79.3	78.0	80.8	81.1	**81.6**
	CS	71.9	75.1	81.7	77.2	84.0	84.1	**85.0**
	RU	72.4	74.6	85.9	77.1	88.1	88.2	**88.6**
	EN	92.9	93.5	94.3	94.2	95.2	95.1	**95.4**
Sem	EN	91.1	91.4	91.4	92.6	93.0	93.1	**93.4**

Table 7: Classification accuracy for combined representations for morphological and semantic tagging. Here, W/B/C stand for word/BPE/character units.

7.2 The Best of Many Worlds

The variations in performance for different representations suggest that they are learning different aspects of language, which might be complementary. Thus, we tried to combine them. Table 7 summarizes the results for morphological and semantic tagging.[9] We can see that combinations involving characters (B+C, W+C in the table) yield

[9]We observed similar trends for the other tasks.

larger improvement compared to combining word- and BPE-based representations (W+B). However, combining all three performed best for all languages and for all tasks.

7.3 State-of-the-Art Embeddings

We connect our findings with recent work on training state-of-the-art embeddings: CoVe (McCann et al., 2017), ELMo (Peters et al., 2018), and BERT (Devlin et al., 2019). Each of these architectures uses different units of representations: e.g., CoVe uses words, BERT is based on subword units, while ELMo focuses on characters.[10]

We speculate that, although these models yield state-of-the-art results for several tasks, their performance may be suboptimal because of the choice of their underlying representation units. In our experiments above, we have shown that it is possible to achieve potentially better performance when using units of different granularity jointly.

We have further shown that the best-performing representation unit is target task-dependent. We believe this would be true for more complex NLP tasks as well. For example, question answering generally requires learning long-range dependencies, and thus embeddings from a character-based model might not be the right choice in this case. Our results show that character-based models are not a viable option for handling long-range dependencies, and subword-based representations might be a better option for such tasks.

7.4 Translation vs. Representation Quality

Table 5 summarizes the translation performance of each system. We can see that in most cases, the subword-based systems perform better than the word-based and the character-based ones. However, this is not true in the case of using their representations as features for a core NLP task as in our experiments above. For example, we have found that character-based representations perform best for the morphological tagging task. On a side note, although BPE-based representations perform better for some tasks, they are sensitive to noise. Their capability of segmenting any OOV word into known subwords may result in less reliable systems. Notably, the translation performance of the BPE-based system can fall below that of the character-based system even in the presence of only 10% noise: from 0.7 BLEU in favor of BPE on clean data to 0.8/1.1 BLEU in favor of characters with synthetic/real errors.

[10]However, note that ELMo uses character convolutions, which is different from a fully character-based model.

8 Conclusion and Future Work

We studied the impact of using different representation units—words, characters, BPE units, and morphological segments—on the representations learned by seq2seq models trained for neural machine translation. In particular, we evaluated the performance of such representations on core natural language processing tasks modeling morphology, syntax, and semantics.

Based on our experiments, we can make the following conclusions:

- Representations derived from subword units are better for modeling syntax.
- Character-based representations are distinctly better for modeling morphology.
- Character-based representations are very robust to noise.
- Using a combination of different representations often works best.

Based on our findings, we further conjecture that although subword-based segmentation based on BPE are a de-facto standard when building state-of-the-art NMT systems, the underlying representations they yield are suboptimal for many external tasks. Character-based representations provide a more viable and robust alternative in this regard, followed by morphological segmentation.

In future work, we plan to study how different units affect representation quality in non-recurrent models such as the Transformer (Vaswani et al., 2017) as well as in convolutional architectures (Gehring et al., 2017). We would also like to explore representations from robustly trained systems, which should improve performance on noisy input (Belinkov and Bisk, 2018; Heigold et al., 2018). Finally, it would be interesting to study representations in other NLP tasks besides neural machine translation.

Acknowledgements

This work was funded by the QCRI, HBKU, as part of the collaboration with the MIT, CSAIL. Yonatan Belinkov was also partly supported by the Harvard Mind, Brain, and Behavior Initiative.

References

Lasha Abzianidze, Johannes Bjerva, Kilian Evang, Hessel Haagsma, Rik van Noord, Pierre Ludmann, Duc-Duy Nguyen, and Johan Bos. 2017. The parallel meaning bank: Towards a multilingual corpus of translations annotated with compositional meaning representations. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, EACL '17, pages 242–247, Valencia, Spain.

Yossi Adi, Einat Kermany, Yonatan Belinkov, Ofer Lavi, and Yoav Goldberg. 2017. Fine-grained analysis of sentence embeddings using auxiliary prediction tasks. In *Proceedings of the International Conference on Learning Representations*, ICLR '17, Toulon, France.

Dzmitry Bahdanau, Jan Chorowski, Dmitriy Serdyuk, Philemon Brakel, and Yoshua Bengio. 2016. End-to-end attention-based large vocabulary speech recognition. In *Proceedings of the 2016 IEEE International Conference on Acoustics, Speech and Signal Processing*, ICASSP '16, pages 4945–4949, Shanghai, China.

Anthony Bau, Yonatan Belinkov, Hassan Sajjad, Nadir Durrani, Fahim Dalvi, and James Glass. 2019. Identifying and controlling important neurons in neural machine translation. In *Proceedings of the International Conference on Learning Representations*, ICLR '19, New Orleans, LA, USA.

Yonatan Belinkov and Yonatan Bisk. 2018. Synthetic and natural noise both break neural machine translation. In *Proceedings of the International Conference on Learning Representations*, ICLR '18, Vancouver, Canada.

Yonatan Belinkov, Nadir Durrani, Fahim Dalvi, Hassan Sajjad, and James Glass. 2017a. What do neural machine translation models learn about morphology? In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, ACL '17, pages 861–872, Vancouver, Canada.

Yonatan Belinkov and James Glass. 2019. Analysis methods in neural language processing: A survey. *Transactions of the Association for Computational Linguistics*, 7:49–72.

Yonatan Belinkov, Lluís Màrquez, Hassan Sajjad, Nadir Durrani, Fahim Dalvi, and James Glass. 2017b. Evaluating layers of representation in neural machine translation on part-of-speech and semantic tagging tasks. In *Proceedings of the 8th International Joint Conference on Natural Language Processing*, IJCNLP '17, pages 1–10, Taipei, Taiwan.

Johannes Bjerva, Barbara Plank, and Johan Bos. 2016. Semantic tagging with deep residual networks. In *Proceedings of the 26th International Conference on Computational Linguistics*, COLING '16, pages 3531–3541, Osaka, Japan.

Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Shujian Huang, Matthias Huck, Philipp Koehn, Qun Liu, Varvara Logacheva, Christof Monz, Matteo Negri, Matt Post, Raphael Rubino, Lucia Specia, and Marco Turchi. 2017. Findings of the 2017 conference on machine translation. In *Proceedings of the Second Conference on Machine Translation*, WMT '17, pages 169–214, Copenhagen, Denmark.

James Bradbury and Richard Socher. 2016. MetaMind neural machine translation system for WMT 2016. In *Proceedings of the First Conference on Machine Translation*, WMT '16, pages 264–267, Berlin, Germany.

Mauro Cettolo, Jan Niehues, Sebastian Stüker, Bentivogli Luisa, and Marcello Federico. 2016. The IWSLT 2016 evaluation campaign. In *Proceedings of the 13th International Workshop on Spoken Language Translation*, IWSLT '16, Seattle, WA, USA.

Junyoung Chung, Kyunghyun Cho, and Yoshua Bengio. 2016. A character-level decoder without explicit segmentation for neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, ACL '16, pages 1693–1703, Berlin, Germany.

Alexis Conneau, Germán Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. 2018. What you can cram into a single $&!#* vector: Probing sentence embeddings for linguistic properties. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, ACL '18, pages 2126–2136, Melbourne, Australia.

Marta R. Costa-jussà and José A. R. Fonollosa. 2016. Character-based neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, ACL '16, pages 357–361, Berlin, Germany.

Fahim Dalvi, Nadir Durrani, Hassan Sajjad, Yonatan Belinkov, D. Anthony Bau, and James Glass. 2019. What is one grain of sand in the desert? Analyzing individual neurons in deep nlp models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, AAAI '19, Honolulu, HI, USA.

Fahim Dalvi, Nadir Durrani, Hassan Sajjad, Yonatan Belinkov, and Stephan Vogel. 2017. Understanding and improving morphological learning in the neural machine translation decoder. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing*, IJCNLP '17, pages 142–151, Taipei, Taiwan.

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 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, NAACL-HLT '19, Minneapolis, MN, USA.

Nadir Durrani, Hassan Sajjad, Hieu Hoang, and Philipp Koehn. 2014. Integrating an unsupervised transliteration model into statistical machine translation. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, EACL '14, pages 148–153, Gothenburg, Sweden.

Javid Ebrahimi, Anyi Rao, Daniel Lowd, and Dejing Dou. 2018. HotFlip: White-box adversarial examples for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, ACL '18, pages 31–36, Melbourne, Australia.

J. Ganesh, Manish Gupta, and Vasudeva Varma. 2017. Interpretation of semantic tweet representations. In *Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2017*, ASONAM '17, pages 95–102, Sydney, Australia.

Ji Gao, Jack Lanchantin, Mary Lou Soffa, and Yanjun Qi. 2018. Black-box generation of adversarial text sequences to evade deep learning classifiers. In *Proceedings of the 2018 IEEE Security and Privacy Workshop*, SPW '18, pages 50–56, San Francisco, CA, USA.

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*, ICML '17, pages 1243–1252, Sydney, Australia.

Ian Goodfellow, Jonathon Shlens, and Christian Szegedy. 2015. Explaining and harnessing adversarial examples. In *Proceedings of the International Conference on Learning Representations*, ICLR '15, San Diego, CA, USA.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition*, CVPR '16, pages 770–778, Las Vegas, CA, USA.

Georg Heigold, Stalin Varanasi, Günter Neumann, and Josef Genabith. 2018. How robust are character-based word embeddings in tagging and MT against wrod scramlbing or randdm nouse? In *Proceedings of the 13th Conference of the Association for Machine Translation in the Americas*, AMTA '18, pages 68–80, Boston, MA, USA.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Julia Hockenmaier. 2006. Creating a CCGbank and a wide-coverage CCG lexicon for German. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, ACL '06, pages 505–512, Sydney, Australia.

Julia Hockenmaier and Mark Steedman. 2007. CCGbank: a corpus of CCG derivations and dependency structures extracted from the Penn Treebank. *Computational Linguistics*, 33(3):355–396.

Robin Jia and Percy Liang. 2017. Adversarial examples for evaluating reading comprehension systems. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, EMNLP '17, pages 2011–2021, Copenhagen, Denmark.

Rekia Kadari, Yu Zhang, Weinan Zhang, and Ting Liu. 2018. CCG supertagging via Bidirectional LSTM-CRF neural architecture. *Neurocomputing*, 283:31–37.

Yoon Kim. 2016. Seq2seq-attn. https://github.com/harvardnlp/seq2seq-attn.

Yoon Kim, Yacine Jernite, David Sontag, and Alexander M. Rush. 2016. Character-aware neural language models. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, AAAI'16, pages 2741–2749, Phoenix, AZ, USA.

Yair Lakretz, German Kruszewski, Theo Desbordes, Dieuwke Hupkes, Stanislas Dehaene, and Marco Baroni. 2019. The emergence of number and syntax units in LSTM language models. In *Proceedings of the 17th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, NAACL-HLT '19, Minneapolis, MN, USA.

Jason Lee, Kyunghyun Cho, and Thomas Hofmann. 2017. Fully character-level neural machine translation without explicit segmentation. *Transactions of the Association for Computational Linguistics*, 5:365–378.

Bin Liang, Hongcheng Li, Miaoqiang Su, Pan Bian, Xirong Li, and Wenchang Shi. 2018. Deep text classification can be fooled. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*, IJCAI '18, pages 4208–4215, Stockholm, Sweden.

Wang Ling, Isabel Trancoso, Chris Dyer, and Alan W. Black. 2015. Character-based neural machine translation. *CoRR*, abs/1511.04586.

Tal Linzen, Emmanuel Dupoux, and Yoav Goldberg. 2016. Assessing the Ability of LSTMs to Learn Syntax-Sensitive Dependencies. *Transactions of the Association for Computational Linguistics*, 4:521–535.

Bryan McCann, James Bradbury, Caiming Xiong, and Richard Socher. 2017. Learned in translation: Contextualized word vectors. In *Proceedings of the Annual Conference on Neural Information Processing Systems: Advances in Neural Information Processing Systems 30*, NIPS '17, pages 6297–6308, Long Beach, CA, USA.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. In *Proceedings of the ICLR Workshop*, Scottsdale, AZ, USA.

Tomas Mikolov, Edouard Grave, Piotr Bojanowski, Christian Puhrsch, and Armand Joulin. 2018. Advances in pre-training distributed word representations. In *Proceedings of the International Conference on Language Resources and Evaluation*, LREC '18, Miyazaki, Japan.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, NIPS '13, pages 3111–3119, Stateline, NV, USA.

Dat Quoc Nguyen, Dai Quoc Nguyen, Dang Duc Pham, and Son Bao Pham. 2014. RDRPOSTagger: A ripple down rules-based part-of-speech tagger. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, EACL '14, pages 17–20, Gothenburg, Sweden.

Nicolas Papernot, Patrick McDaniel, Ananthram Swami, and Richard Harang. 2016. Crafting adversarial input sequences for recurrent neural networks. In *Proceedings of the IEEE Military Communications Conference*, MILCOM '16, pages 49–54, Baltimore, MD, USA.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the Association for Computational Linguistics*, ACL '02, pages 311–318, Philadelphia, PA, USA.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '14, pages 1532–1543, Doha, Qatar.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, NAACL '18, pages 2227–2237, New Orleans, LA, USA.

Peng Qian, Xipeng Qiu, and Xuanjing Huang. 2016a. Analyzing linguistic knowledge in sequential model of sentence. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, EMNLP '16, pages 826–835, Austin, TX, USA.

Peng Qian, Xipeng Qiu, and Xuanjing Huang. 2016b. Investigating language universal and specific properties in word embeddings. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, ACL '16, pages 1478–1488, Berlin, Germany.

Graham Rawlinson. 1976. *The significance of letter position in word recognition*. Ph.D. thesis, University of Nottingham.

Hassan Sajjad, Fahim Dalvi, Nadir Durrani, Ahmed Abdelali, Yonatan Belinkov, and Stephan Vogel. 2017. Challenging language-dependent segmentation for Arabic: An application to machine translation and part-of-speech tagging. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, ACL '17, pages 601–607, Vancouver, Canada.

Suranjana Samanta and Sameep Mehta. 2017. Towards crafting text adversarial samples. *arXiv preprint arXiv:1707.02812*.

Rico Sennrich. 2017. How grammatical is character-level neural machine translation? Assessing MT quality with contrastive translation pairs. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, EACL '17, pages 376–382, Valencia, Spain.

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*, ACL '16, pages 1715–1725, Berlin, Germany.

Xing Shi, Inkit Padhi, and Kevin Knight. 2016. Does string-based neural MT learn source syntax? In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, EMNLP '16, pages 1526–1534, Austin, TX, USA.

Peter Smit, Sami Virpioja, Stig-Arne Grönroos, and Mikko Kurimo. 2014. Morfessor 2.0: Toolkit for statistical morphological segmentation. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, EACL '14, pages 21–24, Gothenburg, Sweden.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. 2014. Intriguing properties of neural networks. In *Proceedings of the International Conference on Learning Representations*, ICLR '14, Banff, Canada.

Ke Tran, Arianna Bisazza, and Christof Monz. 2018. The importance of being recurrent for modeling hierarchical structure. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, EMNLP '18, pages 4731–4736, Brussels, Belgium.

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 30*, NIPS '17, pages 5998–6008. Long Beach, CA, USA.

Ekaterina Vylomova, Trevor Cohn, Xuanli He, and Gholamreza Haffari. 2017. Word representation models for morphologically rich languages in neural machine translation. In *Proceedings of the First Workshop on Subword and Character Level Models in NLP*, SWCN '17, pages 103–108, Copenhagen, Denmark.

Yu-Hsuan Wang, Cheng-Tao Chung, and Hung-yi Lee. 2017. Gate activation signal analysis for gated recurrent neural networks and its correlation with phoneme boundaries. In *Proceedings of the 18th Annual Conference of the International Speech Communication Association*, Interspeech '2017, pages 3822–3826, Stockholm, Sweden.

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, ukasz 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.

Zhizheng Wu and Simon King. 2016. Investigating gated recurrent networks for speech synthesis. In *Proceedings of the 2016 IEEE International Conference on Acoustics, Speech and Signal Processing*, ICASSP '16, pages 5140–5144, Shanghai, China.

Wenduan Xu. 2016. LSTM shift-reduce CCG parsing. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, EMNLP '16, pages 1754–1764, Austin, TX, USA.