WHAT IS IN A TRANSLATION UNIT? COMPARING CHARACTER AND SUBWORD REPRESENTATIONS BEYOND TRANSLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Recent work has shown that contextualized word representations derived from neural machine translation (NMT) 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 syntax, semantics, and morphology. 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 rise 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., 2015), 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, e.g., the ElMo model learns contextualized word embeddings from a language modeling task, using LSTMs (Peters et al., 2018).

More recently, it has been shown that complex tasks such as neural machine translation can yield superior representations (McCann et al., 2017). This is because the internal understanding of the input language that needs to be built by the network in order 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. Such representations have yielded state-of-the-art results for tasks such as sentiment analysis, textual entailment, and question answering.

Unfortunately, computational and memory limitations as of present prevent neural machine translation (NMT) models from using large-scale vocabularies, typically limiting them 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, respectively. The problem is typically addressed using byte-pair encoding (BPE), where words are segmented into pseudo-word character sequences based on frequency (Sennrich et al., 2016). A somewhat less popular solution is to use characters as the basic unit of representation (Chung et al., 2016; Lee et al., 2017). In the case of morphologically complex languages, another alternative is to reduce the vocabulary by using unsupervised morpheme segmentation (Bradbury & 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 & 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 can be summarized 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 make practical recommendations based on our results.

We found that while representations derived from morphological segments are better for modeling syntax, character-based ones are superior for morphology and are also more robust to noise.

2 Related Work

Representation analysis aims at demystifying what is learned inside the neural network black-box. This includes analyzing word and sentence embeddings (Adi et al., 2016; 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 & King, 2016; Wang et al., 2017), and NMT representations (Shi et al., 2016; Belinkov et al., 2017), as applied to morphological (Qian et al., 2016b; Vylomova et al., 2016), semantic (Qian et al., 2016b) and syntactic (Linzen et al., 2016; Tran et al., 2018; Conneau et al., 2018) tasks. While previous work focused on words, here we compare units of different granularities.

Subword translation units aim at reducing vocabulary size and OOV rate. NMT researchers have used BPE units (Sennrich et al., 2016), morphological segmentation (Bradbury & Socher, 2016), characters (Lee et al., 2017), and hybrid units (Ling et al., 2015; Costa-jussà & Fonollosa, 2016). There have also been comparisons between subword units in the context of NMT (Sennrich, 2017). Unlike this work, here we focus on representation learning rather than on translation quality.

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

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 LSTM hidden states from each layer of the encoder/decoder. We concatenate the representations of layers and we use them as feature vector $\mathbf{z}_i$ for the auxiliary task. We train a logistic regression classifier by 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 $\mathbf{l}$.

The weights $\theta \in \mathbb{R}^{D \times L}$ are learned with gradient descent. Here D is the dimensionality of the latent representations $\mathbf{z}_i$ and L is the size of the label set for $\mathcal{P}$.

Table 1: Example sentence with different segmentations.

Words	Professor admits to shooting girlfriend
BPE	Professor admits to sho@@ oting gir@@ l@@ friend
Morfessor	Professor admit@@ s to shoot@@ ing girl@@ friend
Characters	P r o f e s s o r _ 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 sentence with different annotations.

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

3.1 Word Representation Units

We consider four representation units: words, byte-pair encoding (BPE) units, morphological units, and characters. Table 1 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-gram sequences merge to form one symbol. The number of merge operations is controlled by a hyper-parameter `OP`, which directly affects the granularity of segmentation: a high value of `OP` means coarse segmentation and a low value means fine-grained segmentation. 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` *shooting* into base verb *shoot* and the suffix *ing*. Compare this to the BPE segmentation *sho* + *oting*, which has no linguistic justification. On the extreme, the fully *character-level* units treat each word as a sequence of characters.

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 property. We consider two simple approximations to estimate a word representation from subword representations:

(*i*) **Average**: for each word, 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.

(*ii*) **Last**: 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 character- and subword-based representations at the word level via prediction tasks. Such kind of analysis has not been performed before.

3.2 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).

[1] Belinkov et al. (2017) analyzed representations from a charCNN (Kim et al., 2015), but the extracted features were still based on word representations produced by the charCNN. As a result, they could not analyze BPE and character-based models that do not assume segmentation into words.

Table 3: Statistics for NMT and Classifier Training – English (en), German (de), Russian (ru), and Czech (cs) – CV = Cross Validation

(a) NMT Data

	de-en	cs-en	ru-en
Train	507K	340K	370K
Dev	3000	3000	2818
Test	3000	3000	2818

(b) Number of tags

	de	cs	ru	en
Morphology	509	1004	602	42
Semantics	69	–	–	66
Syntax	1272	–	–	–

(c) Classifier Data

	de	cs	ru	en
	Morphology			
Train	14498	14498	11824	14498
Test	8172	8172	6000	8172
	Semantics			
Train	–	–	–	14084
Test	–	–	–	12168
CV	1863	–	–	–
	Syntax			
Train	–	–	–	41586
Test	–	–	–	2407

In particular, we experiment with *morphological tagging* for German, Czech, Russian and English[2] languages, *lexical semantics tagging* for English and German languages, and *syntactic tagging* via CCG supertagging for English language. Table 2 shows an example sentence with annotations of each task. The morphological tags capture word structure, semantic tags show semantic property, and syntax tags (CCG super tags) captures global syntactic information locally at the lexical level. For example in Table 2, – the morphological tag `VBZ` for the word "receives", marks that it is a verb with non-third person singular present property, the semantic tag `ENS` describes a `present simple event` category, and the syntactic tag `S[dcl]\NP)/NP` indicates that the preposition "in" attaches to the verb.

3.3 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 & Bisk (2018). The errors contain a good mix of typos, misspellings, and other kinds of errors. In addition, we create data with synthetic noise. We induced two kinds of errors: i) *Swap* and *Middle*. *Swap* is a common error which occurs when neighboring characters are mistakenly swapped (e.g., word $\rightarrow$ 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$\rightarrow$ eaxmlpe). We corrupt (using *swap* or *middle*) or replace (using real errors corpora) n% words randomly in each test sentence. We then re-extract feature vectors for the erroneous words in a sentence and re-evaluate the prediction capability of these embeddings on the linguistic tasks.

4 Experimental Setup

Data and Languages: We trained NMT systems for 4 language pairs: German-English, Czech-English, Russian-English and English-German, using data made available through the two popular machine translation campaigns, namely, WMT (Bojar et al., 2017) and IWSLT (Cettolo et al., 2016). The MT models were trained using a concatenation of NEWS and TED training data. We used official TED testsets (testsets-11-13) to report translation quality (Papineni et al., 2002). The morphological classifiers were trained and tested on a concatenation of NEWS and TED testsets, which were automatically tagged as described in the next section. Semantic and syntactic classifiers were trained and tested on existing annotated corpora. Statistics are shown in Table 3.

Taggers: We used RDRPOST (Nguyen et al., 2014) to annotate data for the classifier. For semantic tagging, we used the the Groningen Parallel Meaning Bank (Abzianidze et al., 2017). The tags

[2]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.

Table 4: BLEU scores across language pairs.

	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
morfessor → bpe	35.5	28.5	22.5	29.9
char → bpe	34.9	29.0	21.3	30.0

Table 5: OOV rate (%) in the MT and classifier tests.

	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

are grouped in coarse categories such as events, names, time, and logical expressions. Only 1863 annotated sentences (12783 tokens) were available for German. We performed cross-fold evaluation to train and report semantic classification results for German. For CCG supertagging, we used the English CCGBank (Hockenmaier & Steedman, 2007).[3] See Table 3 for statistics.

MT Systems and Classifiers: We used `seq2seq-attn` (Kim, 2016) to train 2-layered attentional long short-term memory (LSTM) (Hochreiter & Schmidhuber, 1997) encoder-decoder systems with bidirectional encoder. We used 500 dimensions for both word embeddings and LSTM states. We trained systems with SGD for 20 epochs and used the final model for generating features for the classifier. We trained the systems in both *-to-English and English-to-* directions and analyze the representations from both *encoder* and *decoder*. To analyze the encoder-side, we fix the decoder-side with BPE-based embeddings and train the source-side with word/BPE/Morfessor/char units. Similarly, to analyze the decoder-side, we train the encoder representation with BPE units and vary the decoder side with different input units. Our motivation for this setup is to analyze representations in isolation keeping the other half of the network static across different settings. We use 50k BPE operations and limit the vocabulary of all systems to 50k. The word/BPE/Morfessor/character-based systems were trained with sentence lengths of 80/100/100/400, respectively.

The classifier is a logistic regression whose input is either hidden states in word-based models, or **Last** or **Average** representations in character- and subword-based models. Since we concatenate forward and backward states from all layers, this ends up being 2000/1000 dimensions when classifying the encoder/decoder: 500 dimensions×2 layers×2 directions (1 for decoder). The classifiers are trained for 10 epochs.

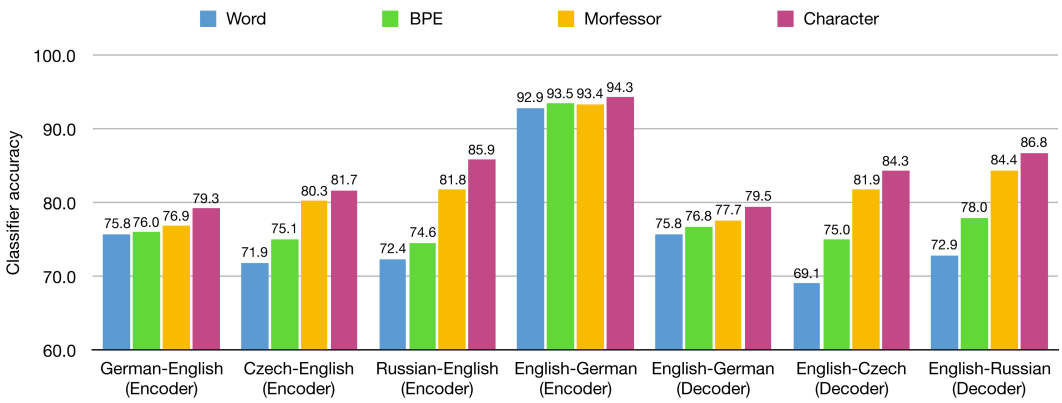


Figure 1: Classifier accuracy on morphological tagging for the various systems and language pairs. The encoder models are trained with BPE as target and the decoder models with BPE as a source.

5 Results

We now present the results of using representations learned from different input units on the task of predicting morphology, semantics and syntax. For subword and character units, we found that the activation of the last subword/character unit of a word consistently better than using the average of all activations. So we present the results using **Last** method only and discuss this more later.

[3]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).

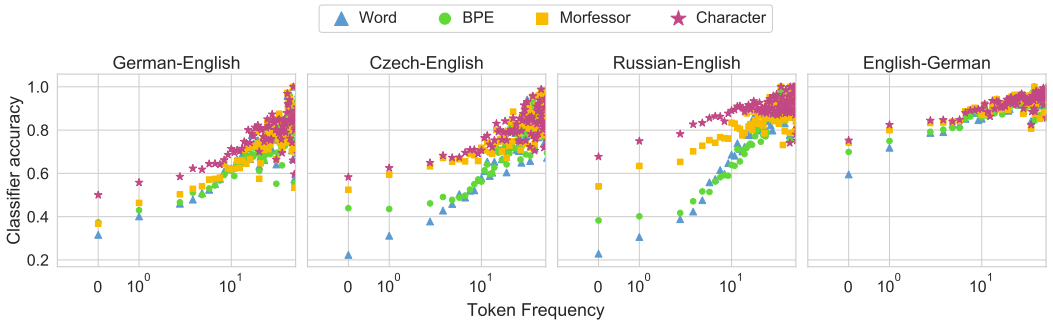
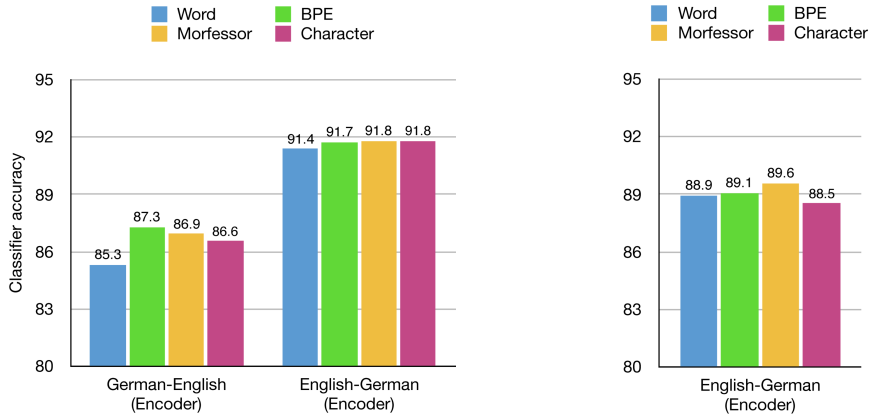


Figure 2: Morphological tagging accuracy vs. word frequency for different translation units.



(a) Semantic tagging

(b) Syntactic tagging

Figure 3: Semantic and syntactic tagging for different units in English (EN) and German (DE).

5.1 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 representations achieved the lowest accuracy. The differences are more significant in the case of languages with relatively complex morphology, Czech and Russian. We see a difference of up to 14% in favor of using character-based representations when compared with the word-based representations. The improvement is minimal in the case of English (1.2%), which is a morphologically simpler language. Comparing subword units as obtained using Morfessor and BPE, we found Morfessor to give much better morphological tagging performance especially in the case of morphologically rich languages, Czech and Russian. This is due to the linguistically motivated segmentations which are helpful in for learning language morphology.

We further investigated whether the performance difference between various representation is due to the difference in modeling infrequent and out-of-vocabulary words. Table 5 shows the OOVs rate of each language which is higher for morphologically rich languages. Figure 2 shows that the gap between different representations is inversely related to the frequency of the word in the training data: character-based models perform much better than others on less frequent and OOV words.

Decoder Representations: Next, we used the decoder representations from the English-to-* models. We saw a similar performance trend as in the case of encoder-side representations, character units performed the best while word units performed the worst. Also morphological units performed better than the BPE-based units. Comparing encoder representation with decoder representation, it is interesting to see that in several cases the decoder-side representations performed better than the encoder-side representations, even though they are trained using a uni-directional LSTM only. Since we did not see any difference in trend between encoder and decoder side representations, we only present the encoder side results in the later part of the paper.

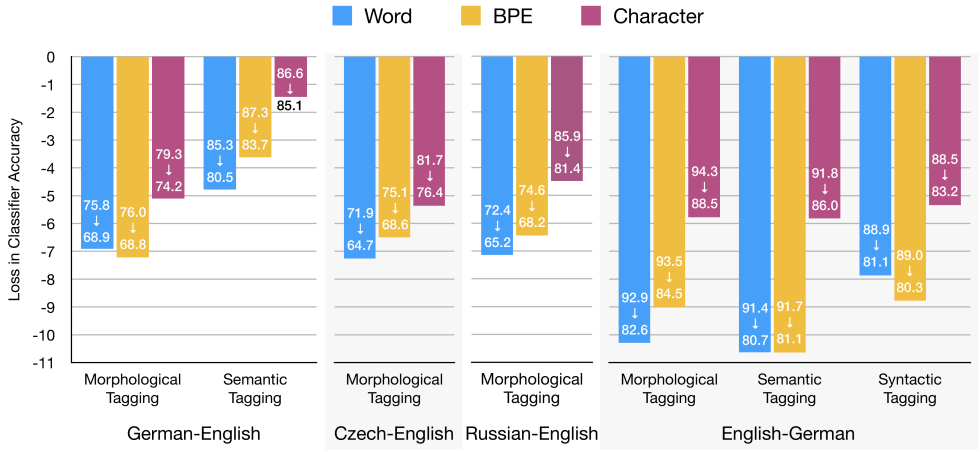


Figure 4: Drop in classification accuracy with 25% noise in each sentence. Absolute scores (original → noisy) are inset. Note that with noisy data, the character-based systems are always better.

5.2 Semantic Tagging

Figure 3 summarizes the results on the semantic tagging task. On English, subword-based (BPE and Morfessor) representations and character-based representation achieve comparable results. However, for German, BPE-based representations performed better than the other representations. These results contrast to morphology prediction results, where character-based representations were consistently better compared to their subword-based counterparts.

5.3 Syntactic Tagging

The final property we evaluate is CCG super-tagging, reflecting syntactic knowledge. Here we only have English tags, so we evaluate encoder representations from English→German models, trained with words, characters, and subwords. We found that morphologically segmented representation units perform the best while words and BPE-based representations perform comparable. The characters-based representations lag behind, though the difference between accuracy is small compared to the morphological tagging results.[4] It is noteworthy that characters perform below both words and subwords here, contrary to their superior performance on the task of morphology. We will return to this point in the discussion in Section 6.

5.4 Robustness to Noise

We now evaluate the robustness of the representations towards noise. We induce errors in the testsets by corrupting 25% of the words in each sentence using different error types (synthetic or real noise), as described in Section 3.3. We extract the representations of the noisy testsets and re-evaluate the classifiers. Figure 4 shows the performance on each task. Evidently, characters yield much better performance on all tasks and for all languages, showing minimal drop in the accuracy, in contrast to earlier results where they did not outperform subword units[5] on the task of syntactic tagging. This result shows character-based representations are more robust towards noise compared to others.

Surprisingly in a few cases, BPE-based representations performed even worst than word-based representations, e.g. in the case of Syntactic tagging (80.3 vs. 81.1). We hypothesize that BPE segments a noisy word into two known subword units that may have no close relationship with the actual word. Using representations of wrong subword units resulted in a significant drop in performance.

We further investigated the robustness of each classifier by increasing the percentage of noise in the test data and found that the difference in representation quality stays constant across BPE and

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

[5]We found similar trends comparing BPE and Morfessor and left out the latter in the interest of space.

Table 6: Morphological tagging performance when combining representations

Language-Pair	Word	BPE	Character	Word+BPE	BPE+Character	Word+Character	ALL
German-English	75.8	76.0	79.3	78.0	80.8	81.1	81.6
Czech-English	71.9	75.1	81.7	77.2	84.0	84.1	85.0
Russian-English	72.4	74.6	85.9	77.1	88.1	88.2	88.6
English-German	92.9	93.5	94.3	94.2	95.2	95.1	95.4

character representations where as word representations deteriorate significantly with increasing amount of noise. Detailed accuracies are included in the supplementary material.

6 DISCUSSION

Comparing Performance Across Tasks Character-based representations outperformed in the case of morphological tagging; BPE-based representations performed better than others in the semantic tagging task for German (and about the same in English); and Morfessor performed slightly better than others for syntax. Syntactic tagging requires knowledge of the complete sentence. Splitting a sentence into characters substantially increases the length (from 50 words in a sentence to 250 characters on average) of the sentence. The character-based models lack in capturing long distance dependencies, which could be a reason for their low performance in this task. Similarly, in case of morphological tagging, the information about the morphology of a word is dependent on the surrounding words plus internal information (root, morphemes etc.) presents in the word. The character-based system has access to all of this information which results in high tagging performance. Morfessor performed better than BPE in the morphological tagging task because its segments are linguistically motivated units (segmented into root + morphemes), making the information about the word morphology explicit in the representation. In comparison, BPE solely focuses on the frequency of characters occurring together in the corpus and can yield linguistically incorrect units.

Translation vs. Representation Quality Table 4 summarizes the translation performance of each system. In most of the cases, the subword-based systems perform better than the word-based and character-based systems. However, this is not true in the case of using their representations as feature in the core NLP tasks. For example, we found that character-based representations perform better than others in the morphological tagging task. On an additional note, BPE-based representations although perform better for some tasks, are sensitive to noise. Their ability to segment any unknown words into two known subwords result in less reliable systems. Notably, the translation performance of the BPE-based system falls below the character-based system even with 10% noise only.

Best of All Worlds The variation in the performance of the representations reflect that they may be learning different aspects of the language. To investigate whether representations are complementary to each other, we train the classifier on their concatenation. Table 6 summarizes the results on the morphological tagging task. The performance of the classifier improved in all combinations of representations while the best results are achieved using all three units together.

7 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 NMT. Unlike previous work, which targeted end tasks such as sentiment analysis and question answering, here we focused on modeling morphology, syntax and semantics. We found that (*i*) while representations derived from subwords units are slightly better for modeling syntax, (*ii*) character representations are distinctly better for modeling morphology, and (*iii*) are also more robust to noise in contrast to subword representations, (*iv*) and that using all representations together works best. Based on our findings, we conjecture that although BPE segmentation is a de-facto standard in building state-of-the-art NMT systems, the underlying representations it yields are suboptimal for 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 explore specialized character-based architectures for NMT. We further want to study how different units affect representation quality in non-recurrent models such as the Transformer (Vaswani et al., 2017) and in convolutional architectures (Gehring et al., 2017).

REFERENCES

Lasha Abzianidze, Johannes Bjerva, Kilian Evang, Hessel Haagsma, Rik van Noord, Pierre Ludmann, Duc-Duy Nguyen, and Johan Bos. 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: Volume 2, Short Papers*, pp. 242–247, Valencia, Spain, April 2017. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/E17-2039.

Yossi Adi, Einat Kermany, Yonatan Belinkov, Ofer Lavi, and Yoav Goldberg. Fine-grained Analysis of Sentence Embeddings Using Auxiliary Prediction Tasks. *arXiv preprint arXiv:1608.04207*, 2016.

Dzmitry Bahdanau, Jan Chorowski, Dmitriy Serdyuk, Philemon Brakel, and Yoshua Bengio. End-to-end attention-based large vocabulary speech recognition. *arXiv preprint arXiv:1508.04395*, 2015.

Yonatan Belinkov and Yonatan Bisk. Synthetic and natural noise both break neural machine translation. In *ICLR, Sixth International Conference on Learning Representations*, Vancouver, Canada, May 2018.

Yonatan Belinkov, Nadir Durrani, Fahim Dalvi, Hassan Sajjad, and James Glass. What do Neural Machine Translation Models Learn about Morphology? In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, Vancouver, Canada, July 2017. Association for Computational Linguistics.

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. Findings of the 2017 conference on machine translation (wmt17). In *Proceedings of the Second Conference on Machine Translation, Volume 2: Shared Task Papers*, pp. 169–214, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/W17-4717.

James Bradbury and Richard Socher. Metamind neural machine translation system for wmt 2016. In *Proceedings of the First Conference on Machine Translation*, pp. 264–267, Berlin, Germany, August 2016. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/W16-2308.

Mauro Cettolo, Jan Niehues, Sebastian Stüker, Bentivogli Luisa, and Marcello Federico. The IWSLT 2016 Evaluation Campaign. In *Proceedings of the 13th International Workshop on Spoken Language Translation (IWSLT 2016)*, Seattle, December 2016.

Junyoung Chung, Kyunghyun Cho, and Yoshua Bengio. A character-level decoder without explicit segmentation for neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1693–1703, Berlin, Germany, August 2016. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/P16-1160.

Alexis Conneau, German Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. 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)*, July 2018.

Marta R. Costa-jussà and José A. R. Fonollosa. Character-based Neural Machine Translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 357–361, Berlin, Germany, August 2016. Association for Computational Linguistics. URL http://anthology.aclweb.org/P16-2058.

Javid Ebrahimi, Anyi Rao, Daniel Lowd, and Dejing Dou. HotFlip: White-Box Adversarial Examples for NLP. *arXiv preprint arXiv:1712.06751*, 2017.

J. Ganesh, Manish Gupta, and Vasudeva Varma. 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, pp. 95–102, New York, NY, USA, 2017. ACM. ISBN 978-1-4503-4993-2. doi: 10.1145/3110025.3110083. URL http://doi.acm.org/10.1145/3110025.3110083.

Ji Gao, Jack Lanchantin, Mary Lou Soffa, and Yanjun Qi. Black-box Generation of Adversarial Text Sequences to Evade Deep Learning Classifiers. *arXiv preprint arXiv:1801.04354*, 2018.

Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N. Dauphin. Convolutional Sequence to Sequence Learning. In Doina Precup and Yee Whye Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1243–1252, International Convention Centre, Sydney, Australia, 06–11 Aug 2017. PMLR. URL http://proceedings.mlr.press/v70/gehring17a.html.

Ian Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*, 2015.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.

Georg Heigold, Stalin Varanasi, Günter Neumann, and Josef Genabith. 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 (Volume 1: Research Papers)*, pp. 68–80. Association for Machine Translation in the Americas, 2018. URL http://aclweb.org/anthology/W18-1807.

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

Julia Hockenmaier. Creating a CCGbank and a wide-coverage CCG lexicon for German. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pp. 505–512. Association for Computational Linguistics, 2006.

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

Robin Jia and Percy Liang. Adversarial Examples for Evaluating Reading Comprehension Systems. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 2011–2021, Copenhagen, Denmark, September 2017.

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

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

Yoon Kim, Yacine Jernite, David Sontag, and Alexander M Rush. Character-aware Neural Language Models. *arXiv preprint arXiv:1508.06615*, 2015.

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

Bin Liang, Hongcheng Li, Miaoqiang Su, Pan Bian, Xirong Li, and Wenchang Shi. Deep Text Classification Can be Fooled. *arXiv preprint arXiv:1704.08006*, 2017.

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

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

Bryan McCann, James Bradbury, Caiming Xiong, and Richard Socher. 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, pp. 6297–6308, Long Beach, CA, USA, 2017.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space. In *Proceedings of Workshop at ICLR*, 2013a.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed Representations of Words and Phrases and their Compositionality. In *Advances in neural information processing systems*, pp. 3111–3119, 2013b.

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

Dat Quoc Nguyen, Dai Quoc Nguyen, Dang Duc Pham, and Son Bao Pham. Rdrpostagger: A ripple down rules-based part-of-speech tagger. In *Proceedings of the Demonstrations at the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 17–20, Gothenburg, Sweden, April 2014. Association for Computational Linguistics. URL `http://www.aclweb.org/anthology/E14-2005`.

Nicolas Papernot, Patrick McDaniel, Ananthram Swami, and Richard Harang. Crafting Adversarial Input Sequences for Recurrent Neural Networks. In *Military Communications Conference, MILCOM 2016-2016 IEEE*, pp. 49–54. IEEE, 2016.

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

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532–1543, 2014. URL `http://www.aclweb.org/anthology/D14-1162`.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In *Proc. of NAACL*, 2018.

Peng Qian, Xipeng Qiu, and Xuanjing Huang. Analyzing Linguistic Knowledge in Sequential Model of Sentence. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 826–835, Austin, Texas, November 2016a. Association for Computational Linguistics. URL `https://aclweb.org/anthology/D16-1079`.

Peng Qian, Xipeng Qiu, and Xuanjing Huang. Investigating Language Universal and Specific Properties in Word Embeddings. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1478–1488, Berlin, Germany, August 2016b. Association for Computational Linguistics. URL `http://www.aclweb.org/anthology/P16-1140`.

Graham Rawlinson. *The significance of letter position in word recognition*. PhD thesis, University of Nottingham, The address of the publisher, 7 1976. An optional note.

Suranjana Samanta and Sameep Mehta. Towards Crafting Text Adversarial Samples. *arXiv preprint arXiv:1707.02812*, 2017.

Rico Sennrich. 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: Volume 2, Short Papers*, pp. 376–382, Valencia, Spain, April 2017. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural Machine Translation of Rare Words with Subword Units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1715–1725, Berlin, Germany, August 2016. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/P16-1162.

Xing Shi, Inkit Padhi, and Kevin Knight. Does String-Based Neural MT Learn Source Syntax? In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 1526–1534, Austin, Texas, November 2016. Association for Computational Linguistics. URL https://aclweb.org/anthology/D16-1159.

Peter Smit, Sami Virpioja, Stig-Arne Grönroos, and Mikko Kurimo. Morfessor 2.0: Toolkit for statistical morphological segmentation. In *Proceedings of the Demonstrations at the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 21–24, Gothenburg, Sweden, April 2014. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/E14-2006.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *International Conference on Learning Representations (ICLR)*, 2014.

Ke Tran, Arianna Bisazza, and Christof Monz. The Importance of Being Recurrent for Modeling Hierarchical Structure. *arXiv preprint arXiv:1803.03585*, 2018.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is All you Need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 30*, pp. 5998–6008. Curran Associates, Inc., 2017. URL http://papers.nips.cc/paper/7181-attention-is-all-you-need.pdf.

Ekaterina Vylomova, Trevor Cohn, Xuanli He, and Gholamreza Haffari. Word Representation Models for Morphologically Rich Languages in Neural Machine Translation. *arXiv preprint arXiv:1606.04217*, 2016.

Yu-Hsuan Wang, Cheng-Tao Chung, and Hung-yi Lee. Gate Activation Signal Analysis for Gated Recurrent Neural Networks and Its Correlation with Phoneme Boundaries. In *Interspeech 2017*, 2017.

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. Google's neural machine translation system: Bridging the gap between human and machine translation. *CoRR*, abs/1609.08144, 2016. URL http://arxiv.org/abs/1609.08144.

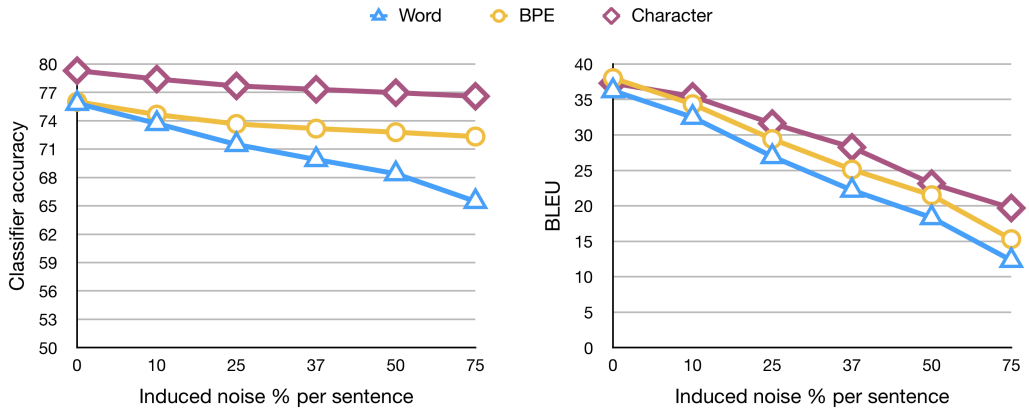
Zhizheng Wu and Simon King. Investigating gated recurrent networks for speech synthesis. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5140–5144. IEEE, 2016.

Wenduan Xu. LSTM shift-reduce CCG parsing. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 1754–1764, 2016.

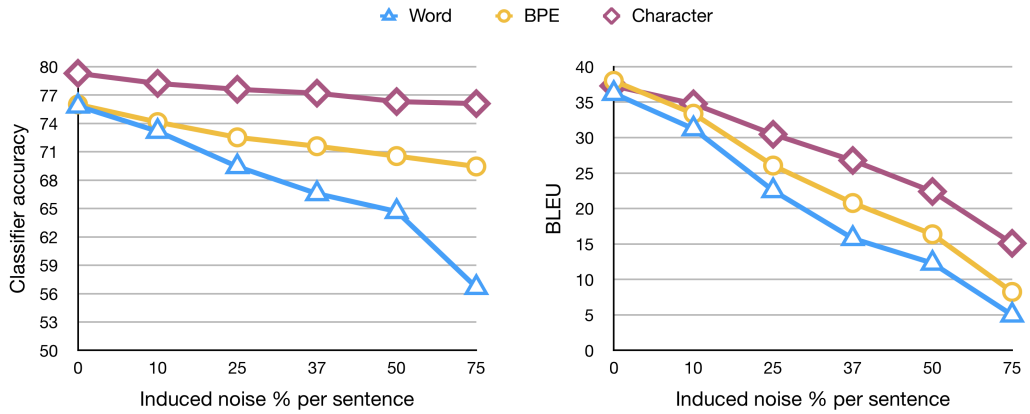
A SUPPLEMENTARY MATERIAL

Table 7: BLEU Scores and Classifier Accuracies (right-most column) on Morphological tagging on original and noisy tests

	tst11	tst12	tst13	Accuracy	tst11	tst12	tst13	Accuracy
	German-to-English				Czech-to-English			
word	36.2	31.1	34.7	75.8	26.4	26.3	29.7	71.9
real	26.9	24.3	26.3	71.5	18.9	20.0	21.5	64.2
swap	22.5	20.3	22.2	69.8	17.3	17.5	19.3	64.6
middle	24.4	21.4	24.4	70.4	18.1	19.3	21.1	65.9
bpe	38.0	32.9	35.9	76.0	27.4	27.0	30.6	75.1
real	29.4	26.1	27.3	73.6	21.8	21.6	23.2	68.6
swap	26.1	23.2	25.3	72.5	18.9	19.0	20.7	71.8
middle	26.9	23.4	25.9	71.9	17.6	18.7	21.5	70.8
char	37.3	32.1	35.2	79.3	27.8	27.7	31.3	81.7
real	31.6	27.7	29.9	77.7	23.1	22.9	25.5	76.3
swap	30.5	26.0	29.4	77.6	23.8	23.2	26.4	79.6
middle	28.0	24.4	26.5	76.4	21.9	22.3	24.9	78.2
	Russian-to-English				English-to-German			
word	21.3	19.8	21.6	72.4	30.3	28.0	30.6	92.9
real	–	–	–	–	15.6	13.7	14.9	82.4
swap	14.8	13.5	15.0	64.7	21.32	17.3	20.2	86.7
middle	16.4	15.2	15.3	65.5	23.1	20.1	21.8	87.7
bpe	22.9	20.9	23.3	74.6	31.5	27.8	31.5	93.5
real	–	–	–	–	17.4	15.0	16.3	86.0
swap	15.5	15.0	16.3	70.4	22.0	18.1	20.6	90.7
middle	16.0	15.0	16.3	69.3	23.1	20.0	22.5	90.4
char	22.1	22.0	21.8	85.9	30.7	28.3	31.1	94.3
real	–	–	–	–	21.8	19.0	20.5	89.5
swap	19.2	17.3	19.0	83.9	26.8	22.9	25.6	93.0
middle	16.9	15.7	15.7	81.2	25.2	22.2	24.4	92.4



(a) Real noise



(b) Synthetic noise

Figure 5: Morphology classification accuracy when increasing amounts of induced noise.