On the Linguistic Representational Power of Neural Machine Translation Models

Yonatan Belinkov*
Massachusetts Institute of Technology
Computer Science and Artificial
Intelligence Laboratory
Harvard University
John F. Paulson
School of Engineering and Applied
Sciences
belinkov@mit.edu

Nadir Durrani*
Qatar Computing Research Institute
HBKU Research Complex
ndurrani@qf.org.qa

Fahim Dalvi
Qatar Computing Research Institute
HBKU Research Complex
faimaduddin@qf.org.qa

Hassan Sajjad
Qatar Computing Research Institute
HBKU Research Complex
hsajjad@qf.org.qa

James Glass
Massachusetts Institute of Technology
Computer Science and Artificial
Intelligence Laboratory
glass@mit.edu

∗ Authors contributed equally.

Submission received: 30 November 2018; revised version received: 21 July 2019; accepted for publication: 17 September 2019.

https://doi.org/10.1162/COLI_a_00367

Despite the recent success of deep neural networks in natural language processing and other spheres of artificial intelligence, their interpretability remains a challenge. We analyze the representations learned by neural machine translation (NMT) models at various levels of granularity and evaluate their quality through relevant extrinsic properties. In particular, we seek answers to the following questions: (i) How accurately is ***word structure*** *captured within the learned representations, which is an important aspect in translating* ***morphologically rich*** *languages? (ii) Do the representations capture long-range dependencies, and effectively handle* ***syntactically divergent*** *languages? (iii) Do the representations capture lexical* ***semantics****? We conduct a thorough investigation along several parameters: (i) Which layers in the architecture capture each of these linguistic phenomena; (ii) How does the choice of translation unit (word, character, or subword unit) impact the linguistic properties captured by the underlying representations? (iii) Do the encoder and decoder learn differently and independently? (iv) Do the representations learned by multilingual NMT models capture the same amount of linguistic information as their bilingual counterparts? Our data-driven, quantitative evaluation illuminates important aspects in NMT models and their ability to capture various linguistic phenomena. We show that deep NMT models trained in an end-to-end fashion, without being provided any direct supervision during the training process, learn a non-trivial amount of linguistic information. Notable findings include the following observations: (i) Word morphology and part-of-speech information are captured at the lower layers of the model; (ii) In contrast, lexical semantics or non-local syntactic and semantic dependencies are better represented at the higher layers of the model; (iii) Representations learned using characters are more informed about word-morphology compared to those learned using subword units; and (iv) Representations learned by multilingual models are richer compared to bilingual models.*

1. Introduction

Deep neural networks have quickly become the predominant approach to most tasks in artificial intelligence, including machine translation (MT). Compared with their traditional counterparts, these models are trained in an end-to-end fashion, providing a simple yet elegant mechanism. This simplicity, however, comes at the price of opaqueness. Unlike traditional systems that contain specialized modules carrying specific sub-tasks, neural MT (NMT) systems train one large network, optimized toward the overall task. For example, non-neural statistical MT systems have sub-components to handle fluency (Heafield 2011), lexical generation (Koehn, Och, and Marcu 2003), word reordering (Galley and Manning 2008; Durrani, Schmid, and Fraser 2011), rich morphology (Koehn and Hoang 2007), and a smorgasbord of features (Chiang, Knight, and Wang 2009) for modeling different phenomena. Neural MT systems, on the other hand, contain a single model based on an encoder-decoder mechanism (Sutskever, Vinyals, and Le 2014) with attention (Bahdanau, Cho, and Bengio 2014). Despite its simplicity, neural MT surpassed non-neural statistical MT within a few years of its emergence. Human evaluation and error analysis revealed that the improvements were obtained through more fluent outputs (Toral and Sánchez-Cartagena 2017) and better handling of morphology and non-local dependencies (Bentivogli et al. 2016). However, it is not clear what the role of different components in the network is, what kind of information is learned during the training process, and how different components interact. Consequently, MT systems trained using neural networks are often thought of as a "black-box"—that is, they map inputs to outputs, but the internal machinery is opaque and difficult to interpret. Gaining a better understanding of these systems is necessary for improving the design

choices and performance. In current practice, their development is often limited to a trial-and-error process, without gaining a real understanding of what the system has learned. We aim to increase model transparency by analyzing the representations learned by NMT models at different levels of granularity in light of various linguistic phenomena—at morphological, syntactic, and semantic levels—that are considered important for the task of machine translation and for learning complex natural language processing (NLP) problems. We thus strive for post-hoc decomposability, in the sense of Lipton (2016). That is, we analyze models after they have been trained, to uncover what linguistic phenomena are captured within the underlying representations. More specifically, we aim to address the following questions in this article:

- What linguistic information is captured in deep learning models?
  - Do the NMT representations capture word morphology?
  - Do the NMT models, being trained on flat sequences of words, still acquire structural information?
  - Do the NMT models learn informative semantic representations?
- Is the language information well distributed across the network or are designated parts (different layers, encoder vs. decoder) more focused on a particular linguistic property?
- What impact does the choice of translation unit (characters, subword units, or words) have on the learned representations in terms of different linguistic phenomena?
- How does translating into different target languages affect the representations on the (encoder) source-side?
- How do the representations acquired by multilingual models compare with those acquired by bilingual models?

To this end, we follow a simple and effective procedure with three steps: (i) train an NMT system; (ii) use the trained model to generate feature representations for source/target language words; and (iii) train a classifier using the generated features to make predictions for a relevant auxiliary task. We then evaluate the quality of the trained classifier on the given task as a proxy to the quality of the trained NMT model. In this way, we obtain a quantitative measure of how well the original NMT system learns features that are relevant to the given task. This procedure has become common for analyzing various neural NLP models (Belinkov and Glass 2019). In this work, we analyze NMT representations through several linguistic annotation tasks: part-of-speech (POS) tagging and morphological tagging for morphological knowledge; combinatory categorial grammar (CCG) supertagging and syntactic dependency labeling for syntactic knowledge; and lexical semantic tagging and semantic dependency labeling for semantic knowledge.

We experiment with several languages with varying degrees of morphological richness and syntactic divergence (compared to English): French, German, Czech, Russian, Arabic, and Hebrew. Our analyses reveal interesting insights such as:

- NMT models trained in an end-to-end fashion learn a non-trivial amount of linguistic information without being provided with direct supervision during the initial training process.

- Linguistic information tends to be organized in a modular manner, whereby different parts of the neural network generate representations with varying amounts and types of linguistic properties.
- A hierarchy of language representations emerges in networks trained on the complex tasks studied in this article. The lower layers of the network focus on local, low-level linguistic properties (morphology, POS, local relations), whereas higher layers are more concerned with global, high level properties (lexical semantics, long-range relations).
- Character-based representations are better for learning morphology, especially for unknown and low-frequency input words. In contrast, representations learned using subword units are better for handling syntactic and semantic dependencies.
- The target language impacts the kind of information learned by the MT system. For example, translating into morphologically poorer languages leads to better source-side word representations. This effect is especially apparent in smaller data regimes.
- Representations learned by multilingual NMT models are richer in terms of learning different linguistic phenomena and benefit from shared learning.[1]

This article is organized into the following sections. Section 2 provides an account of the related work. Section 3 describes the linguistic properties and the representative tasks used to carry out the analysis study. Section 4 describes the methodology taken for analyzing the NMT representations. Section 5 describes data, annotations, and experimental details. Sections 6, 7, and 8 provide empirical results and analysis to evaluate the quality of NMT representations with respect to morphology, syntax, and semantics, respectively, and Section 9 does the same for the multilingual NMT models. Section 10 sheds light on the overall patterns that arise from the experimental results from several angles. Section 11 concludes the article. An open-source implementation of our analysis code is available through the NeuroX toolkit (Dalvi et al. 2019b).

2. Related Work

The work related to this article can be divided into several groups:

2.1 Analysis of Neural Networks

The first group of related work aims at demystifying what information is learned within the neural network black-box. One line of work visualizes hidden unit activations in recurrent neural networks (RNNs) that are trained for a given task (Elman 1991; Karpathy, Johnson, and Li 2015; Kádár, Chrupała, and Alishahi 2017). Although such visualizations illuminate the inner workings of the network, they are often qualitative in nature and somewhat anecdotal. Other work aims to evaluate systems on specific linguistic phenomena represented in so-called challenge sets. Prominent examples include older work on MT evaluation (King and Falkedal 1990), as well as more recent evaluations via contrastive translation pairs (Burlot and Yvon 2017; Rios Gonzales, Mascarell, and

1 The learned parameters are implicitly shared by all the language pairs being modeled.

Sennrich 2017; Sennrich 2017; Bawden et al. 2018). The latter line of work constructs minimal pairs of translations that differ by a known linguistic property, and evaluates whether the MT system assigns a higher score to the correct translation. The challenge set evaluation may produce informative results on the quality of the overall model for some linguistic property, but it does not directly assess the learned representations.

A different approach tries to provide a quantitative analysis by correlating parts of the neural network with linguistic properties, for example, by training a classifier to predict a feature of interest (Adi et al. 2017; Hupkes, Veldhoen, and Zuidema 2017; Conneau et al. 2018). Such an analysis has been conducted on word embeddings (Köhn 2015; Qian, Qiu, and Huang 2016b), sentence embeddings (Adi et al. 2017; Ganesh, Gupta, and Varma 2017; Conneau et al. 2018), and RNN states (Qian, Qiu, and Huang 2016a; Wu and King 2016; Wang, Chung, and Lee 2017). The language properties mainly analyzed are morphological (Qian, Qiu, and Huang 2016b; Vylomova et al. 2016; Belinkov et al. 2017a; Dalvi et al. 2017), semantic (Qian, Qiu, and Huang 2016b; Belinkov et al. 2017b), and syntactic (Tran, Bisazza, and Monz 2018; Köhn 2015; Conneau et al. 2018). Recent studies carried a more fine-grained neuron-level analysis for NMT and LM (Bau et al. 2019a; Dalvi et al. 2019a; Lakretz et al. 2019). In contrast to all of this work, we focus on the representations learned in neural machine translation in light of various linguistic properties (morphological, syntactic, and semantic) and phenomena such as handling low frequency words. Our work is most similar to Shi, Padhi, and Knight (2016) and Vylomova et al. (2016). The former used hidden vectors from a neural MT encoder to predict syntactic properties on the English source side, whereas we study multiple language properties in different languages. Vylomova et al. (2016) analyzed different representations for morphologically rich languages in MT, but they did not directly measure the quality of the learned representations. Surveying the work on analyzing neural networks in NLP is beyond the scope of the present paper. We have highlighted here several of the more relevant studies and refer to Belinkov and Glass (2019) for a recent survey on the topic.

2.2 Subword Units

One of the major challenges in training NMT systems is handling less frequent and out-of-vocabulary words. To address this issue, researchers have resorted to using subword units for training the neural network models. Luong and Manning (2016) trained a hybrid system that integrates character-level representation within a word-based framework. Ling et al. (2015) used a bidirectional long short-term memory network (LSTM; Hochreiter and Schmidhuber 1997) to compose word embeddings from the character embeddings. Costa-jussà and Fonollosa (2016) and Renduchintala et al. (2018) combined convolutional and highway layers to replace the standard lookup-based word representations in NMT systems with character-aware representations.[2] Sennrich, Haddow, and Birch (2016) used **byte-pair encoding** (BPE), a data-compression algorithm, to segment words into smaller units. A variant of this method known as a **wordpiece** model is used by Google (Wu et al. 2016a). Shapiro and Duh (2018) used a similar convolutional architecture on top of BPE. Chung, Cho, and Bengio (2016) used a combination of BPE-based encoder and character-based decoder to improve

2 Character-based systems have been used previously in phrase-based MT for handling morphologically rich (Luong, Nakov, and Kan 2010) and closely related language pairs (Durrani et al. 2010; Nakov and Tiedemann 2012) or for transliterating unknown words (Durrani et al. 2014).

translation quality. Motivated by their findings, Lee, Cho, and Hofmann (2017) explored using fully character representations (with no word boundaries) on both the source and target sides. As BPE segmentation is not linguistically motivated, an alternative to using morpheme-based segmentation has been explored in Bradbury and Socher (2016). It is important to address what using different translation units (word, BPE, morpheme, character) entails. Sennrich (2017) performed a comparative evaluation of character- and BPE-based systems on carefully crafted synthetic tests and found that character-based models are effective in handling unknown words, but perform worse in capturing long-distance dependencies. Our work contributes to this body of research by analyzing how models based on different units capture various linguistic properties. We analyze the representations obtained by training systems on word, character, and BPE-based units.

3. Linguistic Properties

In this section, we describe the linguistic phenomena for which we analyze NMT representations. We focus on linguistic properties that are considered important for the task of machine translation, and that we believe are intrinsically learned in the model to effectively perform the complex task of translation. We consider properties from the realms of morphology, syntax, and semantics. In each case, we describe linguistic properties of interest and define relevant classification tasks that aim to capture them (see Table 1 for sequence labeling tasks).

3.1 Morphology

Modeling the structure of words and their relationship to other words in the sentence is a fundamental task in any NLP application. Languages vary in the way they encode information within words. Some languages exhibit grammatical relations such as subject/object/predicate or gender agreement by only changing the word form, whereas others achieve the same through word order or addition of particles. Morphology (aka word structure), poses an exigent problem in machine translation and is at the heart of dealing with the challenge of data-sparsity. Although English is limited in morphology, other languages such as Czech, Arabic, and Russian have highly inflected morphology. This entails that for each lemma many possible word variants could exist, thus causing an out-of-vocabulary word problem. For example, Huck et al. (2017) found only one morphological variant of the Czech word "čěška" (plural of English "kneecap") in a corpus of 50K parallel sentences. It required 50M sentences, a size of parallel corpus

Table 1
Example sentence with different word-level annotations. The CCG supertags are taken from Nadejde et al. (2017). POS and semantic tags are our own annotation, as well as the German translation and its morphological tags.

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	NP/N	N	(NP\NP) /NP	NP
Words	Obama	empfängt	Netanyahu	in	der	Hauptstadt	der	USA
MORPH	nn.nom. sg.neut	vvfin.3.sg. pres.ind	ne.nom.sg.*	appr.–	art.dat. sg.fem	nn.dat.sg.fem	art.gen.pl.*	ne.gen.pl.*

only available for a handful of language pairs, for them to observe all possible variants of the word. Even if such a data set is available, the computational complexity requires NMT systems to limit the vocabulary size. It is therefore important for an MT system to model word-structure with the available data and vocabulary size limitation. In traditional statistical machine translation, this is often addressed by splitting tokens in morphologically rich languages into constituents in a preprocessing step, using word segmentation in Arabic (Pasha et al. 2014; Abdelali et al. 2016) or compound splitting in German (Koehn and Knight 2003). Previous work also explored generative morphological models, known as **Factored Translation Models**, that explicitly integrate additional linguistic markup at the word level to learn morphology (Koehn and Hoang 2007). In NMT training, using subword units such as byte-pair encoding (Sennrich, Haddow, and Birch 2016) has become a de facto standard in training competition grade systems (Pinnis et al. 2017; Sennrich et al. 2017). A few have tried morpheme-based segmentation (Bradbury and Socher 2016), and several even used character-based systems (Chung, Cho, and Bengio 2016; Lee, Cho, and Hofmann 2017) to achieve similar performance as the BPE-segmented systems.

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*-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. Note that although BPE and Morfessor (unsupervised morpheme-based segmentation) segment words at a similar level of granularity, the segmentation generated by Morfessor (Smit et al. 2014) is linguistically motivated. For example, it splits the gerund verb *shooting* into the base verb *shoot* and the suffix *ing*. In comparison, the BPE segmentation *sho + oting* has no linguistic justification. At the extreme, the fully *character-level* units treat each word as a sequence of characters.

Tagging tasks. In this paper, we study how effective neural MT representations are in learning word morphology and what different translation units offer in this regard. To answer such questions, we focus on the tasks of POS and full morphological tagging, which is the identification of all pertinent morphological features for every word (see Table 1). For example, the morphological tag `vvfin.3.sg.pres.ind` for the word "empfängt" (English 'receives') marks that it is a finite verb, third person, singular gender, present tense, and indicative mood.

Table 2
Example sentence with different segmentations: words, BPE subwords (Sennrich, Haddow, and Birch 2016), Morfessor-based subwords (Smit et al. 2014), and characters. Notice that BPE subwords do not necessarily conform to morphemes ("shooting" → "sho@@" and "oting"), while Morfessor tends to have a more morphological segmentation ("shoot@@", "ing"). "@@" indicates a split subword unit and "_" marks a word boundary.

Words	Professor admits to shooting his girlfriend
BPE	Professor admits to sho@@ oting his gir@@ l@@ friend
Morfessor	Professor admit@@ s to shoot@@ ing his 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 _ h i s _ g i r l f r i e n d

3.2 Syntax

Linguistic theories argue that words are hierarchically organized in syntactic constituents referred to as syntactic trees. It is therefore natural to think that translation models should be based on trees rather than a flat sequence representation of sentences. For more than a decade of research in machine translation, a tremendous amount of effort has been put into syntax-based machine translation (Yamada and Knight (2002); Chiang (2005), Galley et al. (2006), Zhang et al. (2007), Shen, Xu, and Weischedel (2010); Neubig and Duh (2014)), with notable success in languages such as Chinese and German, which are syntactically divergent compared to English. However, the sequence-to-sequence NMT systems were able to surpass the performance of the state-of-the-art syntax-based systems in recent MT competitions (Bojar et al. 2016). The LSTM-based RNN model with the help of the attention mechanism is able to handle long-distance dependencies. There have also been recent attempts to integrate syntax into NMT (Eriguchi, Hashimoto, and Tsuruoka 2016; Stahlberg et al. 2016; Aharoni and Goldberg 2017; Chen et al. 2017; Wu et al. 2017), but sequence-to-sequence NMT models without explicit syntax are the state of the art at the moment (Pinnis et al. 2017; Sennrich et al. 2017).

Tagging tasks. In this paper, we analyze whether NMT models trained on flat sequences acquire structural syntactic information. To answer this, we use two tagging tasks. First, we use CCG supertagging, which captures global syntactic information locally at the word level by assigning a label to each word annotating its syntactic role in the sentence. The process is almost equivalent to parsing (Bangalore and Joshi 1999). For example, the syntactic tag `PP/NP` (in Table 1) can be thought of as a function that takes a noun phrase on the right ("the capital of USA") and returns a prepositional phrase ("in the capital of USA").[3]

Second, we use syntactic dependency labeling, the task of assigning a type to each arc in a syntactic dependency tree. In dependency grammar, sentence structure is represented by a labeled directed graph whose vertices are words and whose edges are relations, or dependencies, between the words (Melčuk 1988; Nivre 2005). A dependency is a directed bi-lexical relation between a head and its dependent, or modifier. Dependency structures are attractive to study for three main reasons. First, dependency formalisms have become increasingly popular in NLP in recent years, and much work has been devoted to developing large annotated data sets for these formalisms. The Universal Dependencies data set (Nivre et al. 2017) that is used in this article has been especially influential. Second, there is a fairly rich history of using dependency structures in machine translation, although much work has focused on using constituency structures (Williams et al. 2016). Third, as dependencies are bi-lexical relations between words, it is straightforward to obtain representations for them from an NMT model. This makes them amenable to the general methodology followed in this paper. Figure 1a shows an example sentence with syntactic dependencies.

3.3 Semantics

The holy grail in MT has long been to achieve an interlingua-based translation model, where the goal is to capture the meaning of the source sentence and generate a target sentence with the same meaning. It has been believed since the inception of MT

3 Refer to Steedman and Baldridge (2011) and Clark and Curran (2004) for more information on CCG supertagging.

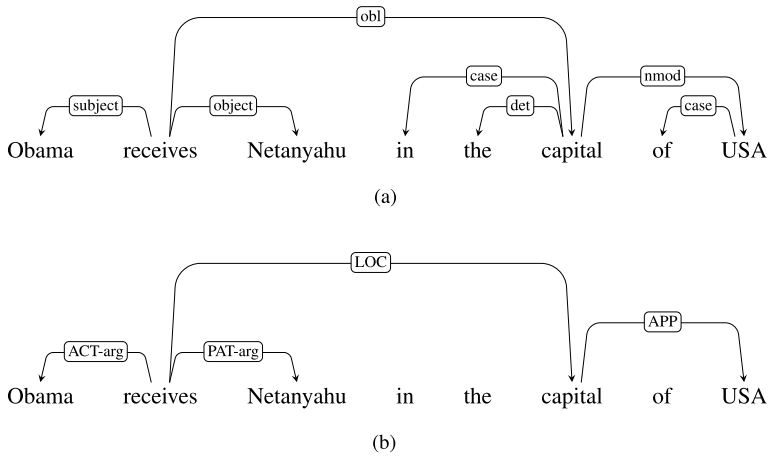


Figure 1
Example sentence with syntactic and semantic relations. (a) Syntactic relations according to the Universal Dependencies formalism. Here "Obama" and " Netanyahu" are the subject and object of "receives", respectively, `obl` refers to an oblique relation of the locative modifier, `nmod` denotes the genitive relation, the prepositions "in" and "of" are treated as case-marking elements, and "the" is a determiner. See `https://universaldependencies.org/guidelines.html` for detailed definitions. (b) Semantic relations according to the PSD formalism. Here `ACT-arg` and `PAT-arg` refer respectively to the originator and affected arguments of "receives", `LOC` in the location, and `APP` is the thing that "capital" belongs to. For detailed definitions, see Cinková et al. (2004).

that without acquiring such meaning representations it will be impossible to generate human-like translations (Weaver 1955). Traditional statistical MT systems are weak at capturing meaning representations (e.g., "who does what to whom—namely, what are the agent, the action, and the patient in the sentence [Jones et al. 2012]). Although neural MT systems are also trained only on parallel data, without providing any direct supervision of word meaning, they are a continuous space model, and are believed to capture word meaning. Johnson et al. (2017), for example, found preliminary evidence that the shared architecture in their multilingual NMT systems learns a universal interlingua. There have also been some recent efforts to incorporate such information in NMT systems, either explicitly (Rios Gonzales, Mascarell, and Sennrich 2017) or implicitly (Liu, Lu, and Neubig 2018).

Tagging task. In this article, we study how semantic information is captured in NMT through two tasks: lexical semantic tagging and semantic dependency labeling. First, we utilize the lexical semantic (SEM) tagging task introduced by Bjerva, Plank, and Bos (2016). It is a sequence labeling task: Given a sentence, the goal is to assign to each word a tag representing a semantic class. This is a good task to use as a starting point for investigating semantics because: (i) tagging words with semantic labels is very simple, compared with building complex relational semantic structures; (ii) it provides a large supervised data set to train on, in contrast to most of the available data sets on word sense disambiguation, lexical substitution, and lexical similarity; and (iii) the proposed SEM tagging task is an abstraction over POS tagging[4] aimed at being language-neutral, and

4 For instance, proximal and distal demonstratives (e.g., "this" and "that") are typically assigned the same POS tag (`DT`) but receive different SEM tags (`PRX` and `DST`, respectively), and proper nouns are disambiguated into several classes such as geo-political entity, location, organization, person, and artifact.

oriented to multilingual semantic parsing, all relevant aspects to machine translation. Table 1 shows an example sentence annotated with SEM tags. The semantic tag ENS describes a present-simple event category.

The second semantic task is semantic dependency labeling, the task of assigning a type to each arc in a semantic dependency graph. Such dependencies are also known as predicate–argument relations, and may be seen as a first step toward semantic structure. They capture different aspects from syntactic relations, as can be noticed by the different graph structure (compare Figure 1b to Figure 1a). Predicate–argument relations have also been used in many (non-neural) MT systems (Komachi, Matsumoto, and Nagata 2006; Wu et al. 2011; Xiong, Zhang, and Li 2012; Li, Resnik, and Daumé III 2013). Figure 1b shows an example sentence annotated with Prague Semantic Dependencies (PSD), a reduction of the tectogrammatical annotation in the Prague Czech–English Dependency Treebank (Cinková et al. 2004; Cinková et al. 2009), which was made available as part of the Semantic Dependency Parsing shared tasks in SemEval (Oepen et al. 2014, 2015).

4. Methodology

We follow a 3-step process for studying linguistic information learned by the trained neural MT systems. The steps include: (i) training a neural MT system; (ii) using the trained model to generate feature representations for words in a language of interest; and (iii) training a classifier using generated features to make predictions for the different linguistic tagging tasks. The quality of the trained classifier on the given task serves as a proxy to the quality of the generated representations. It thus provides a quantitative measure of how well the original MT system learns features that are relevant to the given task.

In this work, we focus on neural MT systems trained using the sequence-to-sequence with attention architecture (Bahdanau, Cho, and Bengio 2014), where an *encoder* network first encodes the source sentence, followed by an *attention* mechanism to compute a weighted average of the encoder states that the *decoder* network uses to generate the target sentence. Both the *encoder* and the *decoder* networks are recurrent neural networks in our case. Several other architectures, for example the Transformer models (Vaswani et al. 2017), have recently been proposed for neural MT. We discuss these briefly in Section 10.5.

Formally, let $s = \{s_1, s_2, \ldots, s_N\}$ denote a source sentence, and $t = \{t_1, t_2, \ldots, t_M\}$ denote a target sentence, where s_i and t_i are words. We first describe the simple case where we have word-level model and linguistic properties. Later we extend this scenario to subword units and to linguistic properties that involve multiple words.

We first use the encoder (Equation 1) to compute a set of hidden states $h = \{h_1, h_2, \ldots, h_N\}$, where h_i represents the hidden state for word s_i. The encoder is a stacked LSTM with L layers, where the output of layer $l-1$ is passed as input to layer l (at each timestep). We then use an attention mechanism to compute a weighted average of these hidden states from the previous decoder state (d_{i-1}), known as the context vector c_i (Equation 2). The context vector is a real valued vector of k dimensions, which is set to be the same as the hidden states in our case. The attention model computes a weight w_{h_i} for each hidden state of the encoder, thus giving soft alignment for each target word. The context vector is then used by the decoder (Equation 3), which is also a stacked LSTM, to generate the next word in the target sequence:

$$\texttt{ENC}_{s_i} : s_i, e_{i-1} \mapsto h_i \qquad (1 \leq i \leq N) \tag{1}$$

$$\texttt{ATTN}_i : \{h_1, \ldots, h_N\}, d_{i-1}, t_{i-1} \mapsto c_i \in \mathbb{R}^k \qquad (1 \leq i \leq M) \tag{2}$$

$$\texttt{DEC}_{t_i} : c_i, d_{i-1}, t_{i-1} \mapsto t_i \qquad (1 \leq i \leq M) \tag{3}$$

After training the NMT system, we freeze the parameters of the network and use the encoder or the decoder as a feature extractor to generate vectors representing words in the sentence. Let $\texttt{ENC}^l_{s_i}$ denote the representation of a source word s_i at layer l in our stacked LSTM. We use $\texttt{ENC}^l_{s_i}$ from a particular layer l or concatenate all layer representations to train the external classifier for predicting a linguistic tag for s_i. The quality of the representation can be deduced from our ability to train a good classifier. For word representations on the target side, we feed our word of interest t_i as the previously predicted word, and extract the representation $\texttt{DEC}_{t_i}$ (see Figure 2 for illustration).

Generating representations for dependency labeling. We used dependency structures to evaluate the syntactic and semantic quality of the learned NMT representations (see Sections 3.2 and 3.3 for details). Given two words that are known to participate in a relation, a classifier is trained to predict the relation type. For the relation labeling task, the input to the classifier is a concatenation of encoder representations for two words in a relation, $\texttt{ENC}^l_{s_i}$ and $\texttt{ENC}^l_{s_j}$, where (s_i, s_j) is a known dependency pair with head s_i and modifier s_j. Again, we perform experiments with both representations from

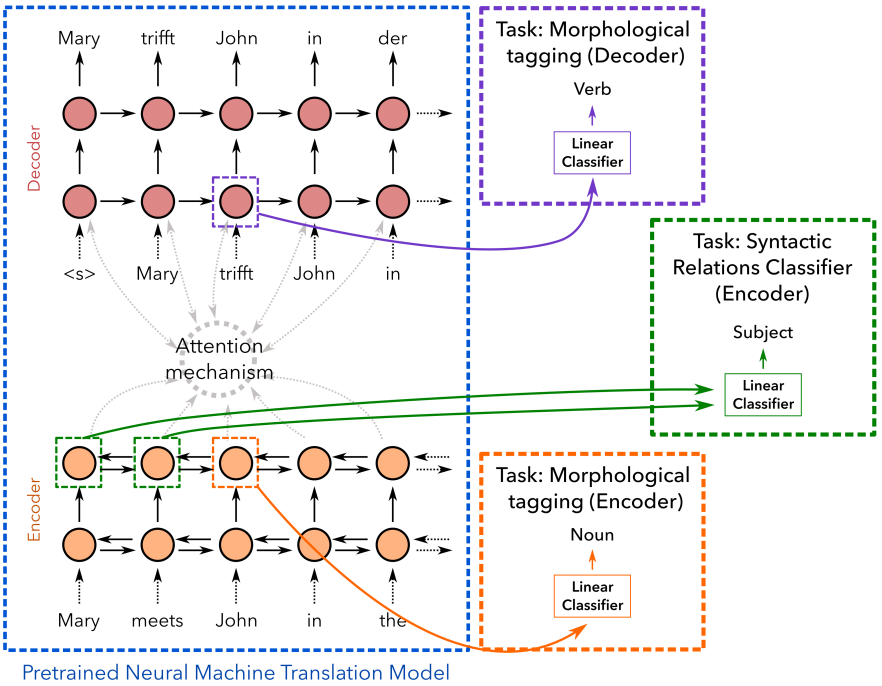


Figure 2
Illustration of our approach: After training an NMT system on parallel data, we extract activations as features from the encoder/decoder and use these along with the labels to train an external classifier. For morphological tagging, we consider the activations for the word alone, while for syntactic/semantic relations we concatenate the activations for the two words involved in the relation.

a particular layer l and the concatenated representation from all layers. Note that this formulation assumes that the order of the dependency is known. This formulation can be seen as a dependency labeling problem, where dependency labels are predicted independently. Although limited in scope, this formulation captures a basic notion of structural relations between words.[5]

Generating representations with subword and character units. Previous work on analyzing NMT representations has been limited to the analysis of word representations only, where there is a one-to-one mapping from translation units (words) and their NMT representations (hidden states) to their linguistic annotations (e.g., POS tags).[6] In the case of character- or BPE-based systems, each word is split into multiple translation units, and each unit has its own representation. It is less trivial to define which representations should be evaluated when predicting a linguistic property such as the part-of-speech. In this work, we consider two simple approximations, illustrated in Figure 3:

1. **Average**: For every source (or target) word, average the activation values of all the subwords (or characters) comprising it. In the case of a bidirectional encoder, we concatenate the averages from the forward and backward encoders' activations on the subwords (or characters) that represent the current word.[7]
2. **Last**: Consider the activation of the last subword (or character) as the representation of the word. For the bidirectional encoder, 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 via prediction tasks, which has not been explored before.

5. Experimental Setup

5.1 NMT Training Data

We experiment with several languages with varying degrees of morphological richness and syntactic divergence (compared to English): Spanish (es), French (fr), German (de), Czech (cs), Arabic (ar), Russian (ru), and Hebrew (he). We trained NMT systems using data made available by 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, TED, and Europarl training data ($\approx$ 2.5M sentence pairs). The multilingual systems were trained by simply concatenating data from different

5 It is also not unrealistic, as dependency parsers often work in two stages, first predicting an unlabeled dependency tree, and then labeling its edges (McDonald and Nivre 2011; McDonald, Lerman, and Pereira 2006). More complicated formulations can be conceived, from predicting the existence of dependencies independently to solving the full parsing task, but dependency labeling is a simple basic task to begin with.

6 Although we studied representations from a charCNN (Kim et al. 2015) in Belinkov et al. (2017a), the extracted features were still based on word representations produced by the charCNN. As a result, in that work we could not analyze and compare subword and character-based models that do not assume a segmentation into words.

7 One could envision more sophisticated averages, such as weighting via an attention mechanism.

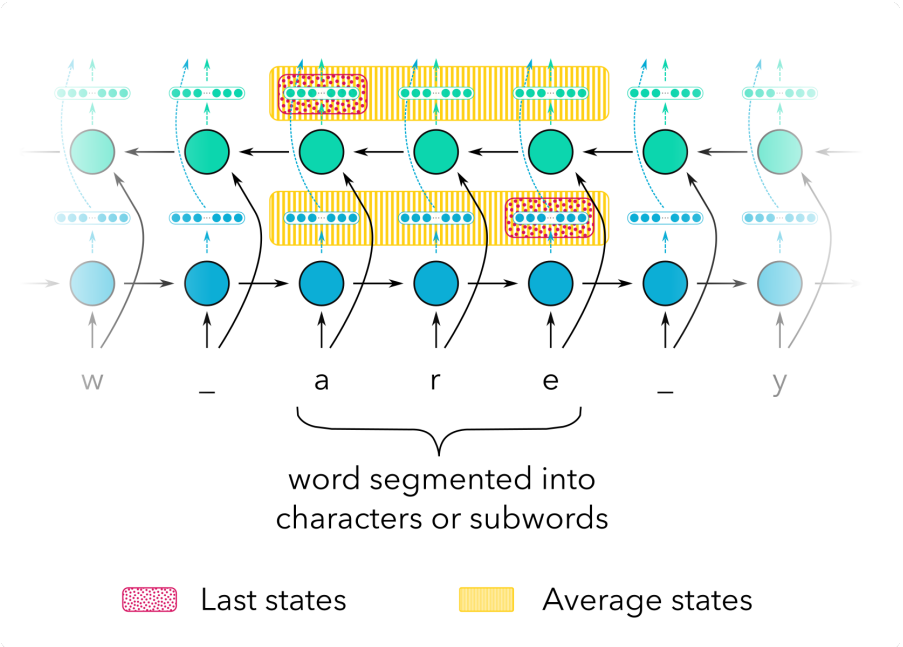


Figure 3
Illustration of a bidirectional layer. Representations from the forward and backward layers are concatenated. For the *average* method, all of the hidden states corresponding to subwords or characters of a given word are averaged together for each layer. For the *last* method, only the hidden state of the final subword or character is considered.

language pairs (a total of ≈10M sentence pairs) and training a shared encoder-decoder pipeline. We used German, French, Spanish, and Czech to/from English to train multilingual systems. Language codes were added as prefixes before each sentence. We used official TED test sets to report translation quality (Papineni et al. 2002). We also used the fully aligned United Nations corpus (Ziemski, Junczys-Dowmunt, and Pouliquen 2016) for training the models in some of our experiments. It includes six languages: Arabic, Chinese, English, French, Spanish, and Russian. This data set has the benefit of multiple alignment of the several languages, which allows for comparable cross-linguistic analysis, for example, studying the effect of only changing the target language. We used the first 2 million sentences of the training set, using the official training/development/test split.

5.2 Neural MT Systems

5.2.1 Preprocessing. We used the standard Moses (Koehn et al. 2007) pipeline for preprocessing the data, which includes tokenization, filtering for length, and true-casing. The systems were trained with a a maximum sentence length of 80 words. For the BPE systems, we used a vocabulary size using 50,000 operations. In the case of multilingual systems, we used 90,000 operations. For the character-based systems, we simply split the words into characters.[8] We used Morfessor (Smit et al. 2014) with default settings to get morpheme-segmented data. The subword (BPE and Morfessor) and character-based

8 We also explored charCNN (Kim et al. 2015; Costa-jussà and Fonollosa 2016) models in our preliminary experiments, and found the charCNN variant to perform poorly, compared with the simple char-based LSTM model both in translation quality and when comparing classifier accuracy. Therefore, we decided to leave them out for brevity. See Appendix for the results.

systems were trained with a maximum sentence length of 100, 100–120, and 400–550 units, respectively.[9]

5.2.2 Model Training. We used the `seq2seq-attn` implementation (Kim 2016) with the following default settings: word embeddings and LSTM states with 500 dimensions, initialized with default `Torch` initialization, SGD with an initial learning rate of 1.0 and decay rate of 0.5 (after the ninth epoch), and dropout rate of 0.3. We used 2–4 hidden layers for both the encoder and the decoder. The NMT system was trained for 20 epochs, and the model with the best validation loss was used for generating features for the external classifier. These are the settings that we have generally used for the experiments reported in this article. We will explicitly mention in the individual sections where we digress from the prescribed settings.

5.3 Classifier Settings

The classifier is a logistic regression model whose input is either hidden states in word-based models, or **Last** or **Average** representations in character- and subword-based models. Because we concatenate forward and backward states from all layers, this ends up being 4,000/2,000 dimensions when classifying the encoder/decoder: 500 dimensions $\times$ 4 layers $\times$ 2 directions (1 for decoder). The objective function is categorical cross-entropy, optimized by Adam (Kingma and Ba 2014). Training is run with shuffled mini-batches of size 512 and stopped after 20 epochs.

The choice of classifier is motivated by two considerations. First, the classifier takes features only from the current word (or word-pair), without additional context. The goal is to evaluate how well the word representation itself captures pertinent information, potentially including contextual information through the NMT LSTM encoder or decoder. Second, using a linear classifier enables focusing on the quality of the representations learned by the NMT system, rather than obtaining state-of-the-art prediction performance. In the literature on analyzing neural representations by classification tasks, simple linear classifiers are a popular choice (Belinkov and Glass 2019). Using a stronger classifier may lead to better overall numbers, but does not typically change the relative quality of different representations (Qian, Qiu, and Huang 2016b; Belinkov 2018, Chapter D.1), which is our main concern in this work.

5.4 Supervised Data and Annotations

We make use of gold-standard annotations wherever available, but in some cases we have to rely on using automatic taggers to obtain the annotations. In particular, to analyze the representations on the decoder side, we require parallel sentences.[10] It is difficult to obtain gold-standard data with parallel sentences, so we rely on automatic annotation tools. An advantage of using automatic annotations, though, is that we can reduce the effect of domain mismatch and high out-of-vocabulary (OOV) rate in analyzing these representations.

We used Tree-Tagger (Schmid 1994) for annotating Russian and the MADAMIRA tagger (Pasha et al. 2014) for annotating Arabic. For the remaining languages (French,

9 The sentence length was varied across different configurations, to keep the training data sizes the same for all systems.

10 We need source sentences to generate encoder states, which in turn are required for obtaining the decoder states that we want to analyze.

Table 3
Train and test data (number of sentences) used to train MT classifiers to predict different tasks. We used automated tools to annotate data for the morphology tasks and gold annotations for syntactic and semantics tasks.

		de	en	cs	ru	fr	es
POS tags	Train	14,498	14,498	14,498	11,824	11,495	14,006
	Test	8,172	8,172	8,172	5,999	3,003	5,640
Morph tags	Train	14,498	14,498	14,498	11,824	11,495	14,006
	Test	8,172	8,172	8,172	5,999	3,003	5,640
CCG tags	Train	–	41,586	–	–	–	–
	Test	–	2,407	–	–	–	–
Syntactic dependency	Train	14,118	12,467	14,553	3,848	–	–
	Test	1,776	4,049	1,894	1,180	–	–
Semantic tags	Train	1,490	14,084	–	–	–	–
	Test	373	12,168	–	–	–	–
Semantic dependency	Train	–	12,000	11,999	–	–	–
	Test	–	9,692	10,010	–	–	–

German, Spanish, and Czech) we used RDRPOST (Nguyen et al. 2014), a state-of-the-art morphological tagger. For experiments using gold tags, we used the Arabic Treebank for Arabic (with the versions and splits described in the MADAMIRA manual) and the Tiger corpus for German.[11] For semantic tagging, we used the semantic tags from the Groningen Parallel Meaning Bank (Abzianidze et al. 2017). For syntactic relation labeling we used the Universal Dependencies data set (Nivre et al. 2017). For CCG supertagging we used the English CCGBank (Hockenmaier and Steedman 2007).[12] For semantic dependency labeling we used PSD, which is a reduction of the tectogrammatical analysis layer of the Prague Czech–English Dependency Treebank, and is made available as part of the Semantic Dependency Parsing data set (Oepen et al. 2014, 2015). Most of the PSD dependency labels mark semantic roles of arguments, which are called functors in the Prague dependency treebank.[13] PSD annotations are available in English and Czech. Table 3 provides the amount of data used to train the MT classifiers for different NLP tasks. Table 4 details the number of tags (or labels) in each task across different languages.

6. Morphology Results

In this section, we investigate what kind of morphological information is captured within NMT models, using the tasks of POS and morphological tagging. To probe this, we annotated a subset of the training data (see Table 3) using POS or morphological

11 `http://www.ims.uni-stuttgart.de/forschung/ressourcen/korpora/tiger.html`.

12 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).

13 The main differences between PSD and the original tectogrammatical annotation are the omission of elided elements, such that all nodes are surface tokens; the inclusion of functional and punctuation tokens; ignoring most cases of function word attachments to content words; ignoring coreference links; and ignoring grammatemes (tectogrammatical correlates of morphological categories). As a side effect, these simplifications make it straightforward to generate representations for surface tokens participating in dependency relations under the PSD formalism. See `http://sdp.delph-in.net` for more information on PSD and refer to Cinková et al. (2009) for details on the original tectogrammatical annotations.

Table 4
Number of tags (for word-level tasks) and labels (for relation-level tasks) per task in different languages.

	de	cs	ru	en	ar	fr	es
POS tags	54	–	–	42	42	33	–
Morphological tags	509	1,004	602	–	1,969	183	212
Semantic tags	69	–	–	66	–	–	
CCG tags	–	–	–	1,272	–	–	
Syntactic dependency labels	35	41	40	–	–	40	32
Semantic dependency labels	–	64	–	87	–	–	–

taggers. We then generated features from the trained NMT models and trained a linear classifier using these features to predict the POS or morphological tags.

Although our goal is not to surpass state-of-the-art tagging performance, we still wanted to compare against several reference points to assess the quality of the underlying representations. To this end we report several baselines: (i) A simple local majority baseline where each word is assigned its most frequent tag and unknown words are assigned the most frequent global tag. (ii) We annotated the data used to train NMT models using the tools mentioned above and trained `char-to-tag` models using the same sequence-to-sequence regime we used to train our MT systems. This can be seen as a skyline reference. (iii) To have a closer comparison with our MT classifier, we generate features from the trained `char-to-tag` models and train a linear classifier using these features. This allows us to exactly compare representations learned for the task of translation versus the representations that are directly optimized toward the task (POS or morphological tagging, for example).

Table 5 shows the prediction accuracy of the classifiers trained on the *encoder-side* representations. MT classifiers always outperform the majority baseline, which entails that the representations contain non-trivial linguistic information about language

Table 5
POS and morphological tagging results: Comparing classifier trained on char-based NMT representations with several baselines: (i) Local majority baseline (most frequent tag), (ii) Character-to-tag trained using sequence-to-sequence model on the same training data as the MT systems, (iii) Classifier trained on representations extracted from (ii) to match the MT generated representations). NMT systems used here to extract representations are character-based models, trained on translating each language to English (and English to German). The classifier results are substantially above the majority baseline, indicating that NMT representations learn non-trivial amounts of morphological information.

		de	cs	ru	en	fr
POS tags	MT classifier	94.0	–	–	95.8	96.3
Baselines	Majority	88.4	–	–	90.1	92.6
	char-to-POS	98.3	–	–	97.7	99.2
	POS classifier	95.4	–	–	96.0	98.5
Morphological tags	MT classifier	80.5	85.2	87.7	–	88.2
Baselines	Majority	68.3	70.4	74.8	–	84.7
	char-to-Morph	92.7	95.7	94.2	–	98.6
	Morph classifier	89.6	90.5	90.5	–	95.8

morphology. The accuracy is high when the language is morphologically poor (e.g., English) or the task is simpler (fewer tags to predict; see Table 4). On the contrary, the accuracy in the case of a morphologically rich language such as Czech is lower. The `char-to-POS/Morph` baselines seems to give much higher numbers compared with ours, but remember that these models are trained on considerably more data (the entire data on which the MT models were trained) and with a more sophisticated bilingual LSTM with attention model, compared with the MT classifier, which is trained on a small subset of neural activations using a simple logistic regression. A much closer skyline reference is the POS/Morph classifiers that are trained on the same data and model architecture as the MT classifier, with the difference that the former is trained on the representations optimized for the task itself whereas the latter is trained on the representations optimized toward the task of machine translation. Therefore, this is still comparing apples to oranges, but provides a more exact reference for the quality of MT representations with respect to learning morphology.

We now proceed with answering more specific questions regarding several aspects of the NMT systems: (i) How do the representations trained from different translation units (word vs. character vs. subword units) compare? (ii) How do the representations trained from the encoder and decoder compare? (iii) What kind of information do different layers capture? and (iv) How does the target language affect the learned source language representations?

6.1 Impact of Translation Unit on Learning Morphology

We trained NMT systems with different translation units: word, character, and subword units, of which we tried two, namely, BPE (Sennrich, Haddow, and Birch 2016) and morphological segmentation (Smit et al. 2014). For subword and character units, we found that the activation of the last subword/character unit of a word performed consistently better than using the average of all activations, so we present the results using the **Last** method throughout the article (see Table 6 for comparison).

Figure 4 summarizes the results of predicting morphology with representations learned by different models. 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, notably 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

Table 6
Classification accuracy for morphological tags using representations generated by aggregating BPE subword or character representations using either the average or the last LSTM state for each word. Here the representations are obtained by concatenating the encoding layers of NMT models trained on translating each language to English. Using the last hidden state consistently outperforms the average state.

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

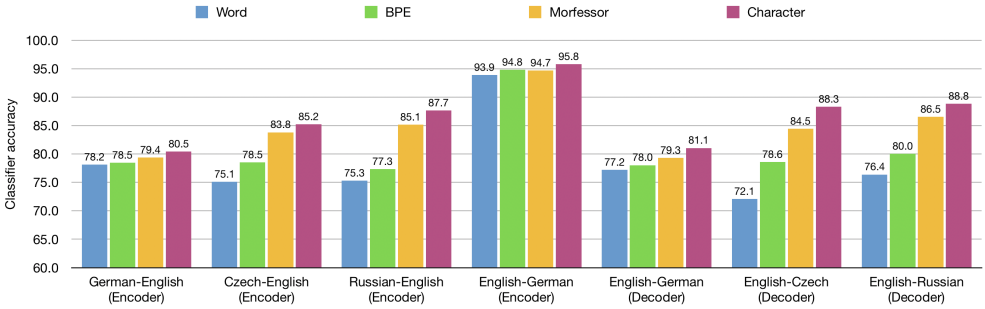


Figure 4
Morphological classification accuracy with different translation units and language pairs. When comparing encoder (decoder) representations, we train NMT models with different source (target) side translation units—words, BPE subwords, Morfessor subwords, or characters—and hold the target (source) side unit fixed as BPE subwords.

Table 7
BLEU scores across language pairs with different translation units on the source side (the target side is held fixed as BPE). The NMT models are trained on NEWS+TED data.

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

Table 8
OOV rate (%) in the (source-side) MT and morphological classification test sets. The morphologically richer Czech (cs) and Russian (ru) have higher OOV rates.

	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

case of English (1.2%), which is a morphologically simpler language. Comparing subword units as obtained using Morfessor and BPE, we found Morfessor to provide much better morphological tagging performance, especially in the case of the morphologically richer languages, Czech and Russian. The representations learned from morpheme-segmented units were found helpful in learning language morphology. These findings are also somewhat reflected in the translation quality (see Table 7). The character-based segmentation gave higher BLEU scores compared with a BPE-based system in the case of the morphologically rich language Czech, but character-based models performed poorly in the case of German, which requires handling long-distance dependencies. Our results (discussed later in Section 7) show that character-based representations are less effective at handling syntactic dependencies.

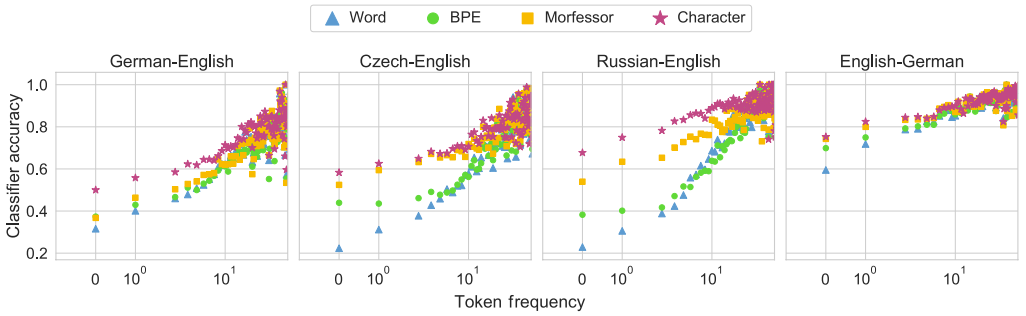


Figure 5
Morphological tagging accuracy vs. word frequency for different translation units on the encoder-side. The target side is held fixed as BPE. The representations for training the morphological classifier are obtained from the top layer of the encoder. Character representations perform better than other ones, especially in low-frequency regimes.

6.1.1 Handling Unknown and Low Frequency Words. We further investigated whether the performance difference between various representations is due to the difference in modeling infrequent and OOV words. As Table 8 shows, the morphologically richer languages have higher OOV rates. Figure 5 reveals 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. The ranking of different units in low frequency regimes is consistent with the overall results in Figure 4—characters perform best, followed by Morfessor subwords, BPE subwords, and words.

6.2 Encoder versus Decoder Representations

The decoder DEC is a crucial part in an MT system with access to both source-side representations and partially generated target-side representations, which it uses to generate the next target word. We now examine whether the representations learned on the decoder-side possess the same amount of morphological knowledge as the encoder side. To probe this, we flipped the language direction and trained NMT systems with English→{German, Czech, Russian} configurations. Then, we use the trained model to encode a source sentence and generate features for words in the target sentence. These features are used to train a classifier on morphological tagging on the target side. Note that in this case the decoder is given the correct target words one by one, similar to the usual NMT training regime. The right-hand side of Figure 4 shows a similar performance trend as in the case of encoder-side representations, with character units performing the best and word units performing the worst. Again, morphological units performed better than the BPE-based units.

Comparing encoder representations with decoder representations, 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 unidirectional LSTM only. Because we did not see any notable trends in differences between encoder and decoder side representations, we only present the encoder-side results in the rest of the paper.

6.3 Effect of Network Depth

Modern NMT systems use very deep architectures (Wu et al. 2016b; Zhou et al. 2016). We are interested in understanding what kind of information different layers capture. Given a trained NMT model with multiple layers, we extract feature representations from the different layers in the encoder. We trained 4-layered models (using (NEW+TED+Europarl data).

Figure 6 shows morphological tagging results using representations from different encoder and decoder layers across five language pairs. The general trend shows that representations from the first layer are better than those from the higher layers, for the purpose of capturing morphology. We found this observation to be true in multi-layered decoder as well (see the right side of Figure 6). We verified these findings with models trained using 2, 3, and 4 layers. Layer 1 was consistently found to give better accuracy on the task of POS tagging and morphology learning. We also found the pattern to hold for representations trained on other units (e.g., character-based units).

Another interesting result to note is that concatenating representations from all the layers gave significantly better results compared to any individual layer (see **Combination** bars in Figure 6). This implies that although much of the information related to morphology is captured at the lower layer, some of it is also distributed to the higher layers. We analyzed individual tags across layers and found that open class categories such as *verbs* and *nouns* are distributed across several layers, although the majority of the learning of these phenomena is still done at layer 1. Please refer to Dalvi et al. (2019a) for further information.

6.4 Effect of Target Language

The task of machine translation involves translating from one language into another. While translating from morphologically rich languages is a challenging task, translating into such languages is even harder. How does the target language affect the learned source language representations? Does translating into a morphologically rich language

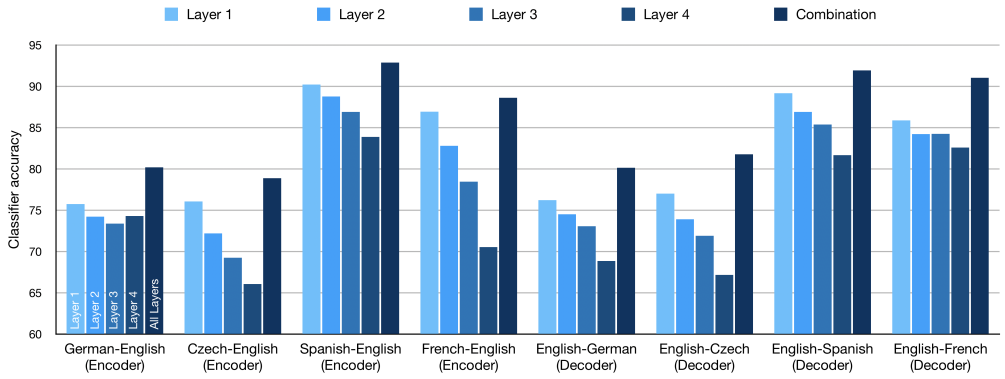


Figure 6
Morphological tagging accuracy using representations from layers 1 to 4, taken from encoders and decoders of different language pairs. Here, the NMT models were trained with BPE units. Layer 1 generates the best representations and in most cases there is a gradual decrease with each layer. The combination of representations improves beyond layer 1, indicating that some morphological information is distributed among layers.

require more knowledge about source language morphology? To investigate these questions, we trained NMT models by keeping the source side constant and using different target languages. To make a fair comparison, the models are trained on the intersection of the training data based on the source language.

Figure 7 shows the results of such an experiment, with models translating from Arabic to several languages: English, Hebrew, German, and Arabic itself. These target languages represent a morphologically poor language (English), a morphologically rich language with similar morphology to the source language (Hebrew), and a morphologically rich language with different morphology (German). As the figure shows, the representations that are learned when translating into English are better for predicting POS or morphology than those learned when translating into German, which are in turn better than those learned when translating into Hebrew.

How should we interpret these results? English is a morphologically poor language that does not display many of the morphological properties that are found in the Arabic source. In contrast, German and Hebrew have richer morphologies, so one could expect that translating into them would make the model learn more about morphology. However, Arabic representations learned from the Arabic→English model are superior in learning morphology. A possible explanation for this phenomenon is that the Arabic→English model is simply better than the Arabic→Hebrew and Arabic-German models, as hinted by the BLEU scores. The inherent difficulty in translating Arabic to Hebrew/German may affect the ability to learn good representations of word structure or perhaps more data is required in the case of these languages to learn Arabic representations of the same quality. However, it turns out that an Arabic→Arabic autoencoder learns to recreate the test sentences extremely well, even though its word representations are actually inferior for the purpose of POS/morphological tagging (Figure 7). This implies that a higher BLEU score does not necessarily entail better morphological representations. In other words, a better translation model learns more informative representations, but only when it is actually learning to translate rather than merely memorizing the data as in the autoencoder case. We found these results to be consistent in other language pairs, that is, by changing the source from Arabic to German and Czech and also using character models instead of words (see Section A.2 in the Appendix for more details); however, more through study is required along this direction as Bisazza and Tump (2018) performed a similar experiment on a fine-grained tag level and found contrastive results.

7. Syntax Results

To evaluate the NMT representations from a syntactic perspective, we consider two tasks. First, we made use of CCG supertagging, which is assumed to capture syntax at the word level. Second, we used dependency relations between any two words in the sentence for which a dependency edge exists, to investigate how words compose. Specifically, we ask the following questions: (i) Do NMT models acquire structural information while they are being trained on flat sequences of bilingual sentences? (ii) How do representations trained on different translation units (word vs. character vs. subword units) compare with respect to syntax? and (iii) Do higher layers learn better representations for these kinds of properties than lower layers?

The analysis carried out previously was chiefly based on lexical properties. To strengthen our analysis, we further used dependency relations that are available for many different language pairs unlike CCG supertags. Here we concatenate the representations of two words in a relation and ask the classifier to predict their syntactic

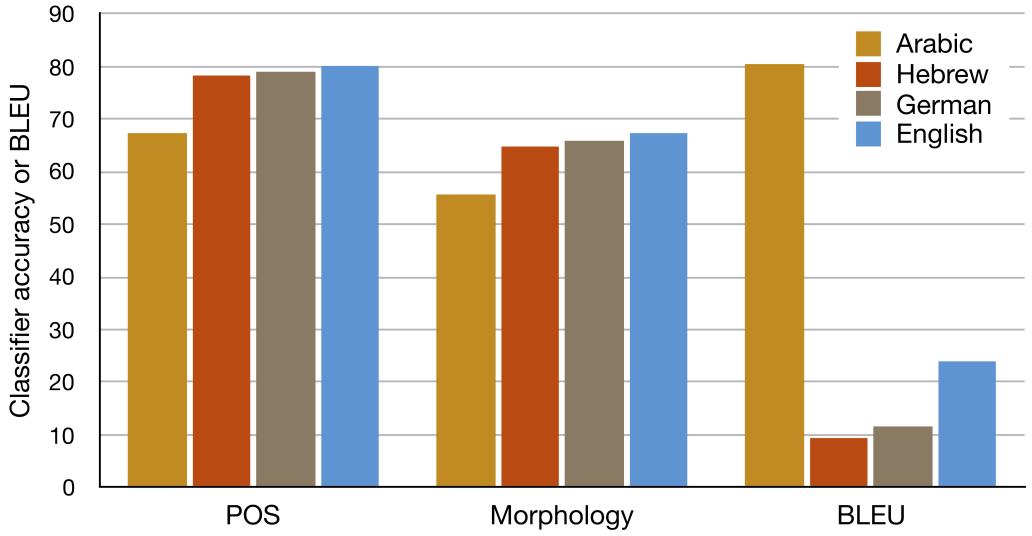


Figure 7
Effect of target language on representation quality of the Arabic source. POS and morphological tagging accuracy, and BLEU scores, using encoder-side representations from NMT models trained with different target languages. These results are obtained with top-layer representations of 2-layer word-based NMT models. The target language has a small effect on source-side representation quality, with better MT models generating better representations, except in the auto-encoder case (Arabic bar).

relation. Table 9 shows that NMT representations are syntax aware. In both tasks (CCG supertagging and syntactic dependency labeling), the classifier accuracy is much higher compared to the local majority baseline,[14] demonstrating that the representations learned during NMT training learn non-trivial amounts of syntactic information.[15] We now proceed to answer the other two questions, namely, the impacts of translation unit and representation depth.

7.1 Impact of Translation Unit on Learning Syntax

Although character-based models are effective at handling unknown and low-frequency words, they have been found poor at capturing long-distance dependencies. Sennrich (2017) performed an evaluation based on contrastive translation pairs and found the subword-based system better in capturing long-distance dependencies. Here

14 For the syntactic dependency majority baseline, we assume the most frequent label of the arc (head-modifier pair). When the pair is unseen during test, we ignore the head and fall back to using modifier only. It is non-trivial to train sequence-to-sequence models for the dependency tasks, so we only rely on the majority baseline for comparison.

15 We do not have similar baselines for the syntax and semantic tasks as we have for the task of morphology prediction. The reason for this discrepancy is that we used automatic tools for annotating data for POS and morphological tagging, but gold annotated data for syntax and semantic tasks. Whereas state-of-the-art POS and morphological tagging tools are freely available, the same is not true for semantic and CCG tagging. We therefore resorted to use the published numbers as the skyline baseline in this case. For syntactic and semantic dependency labeling tasks an additional complexity is how to train a `seq-to-seq` baseline with such annotations. Remember that the task involves modeling head and modifier word to predict a dependency relation. In the case of semantic dependency, there could be multiple heads for a modifier word.

Table 9
Local majority baseline (most frequent tag/label) and classification accuracy using encoder representations generated by the NMT models, on syntactic tasks. The models are trained on translating each language to English (or German in the English case). The classifier results are far superior to the majority baseline, indicating that NMT representations contain a non-trivial amount of syntactic information.

		de	cs	ru	en	fr
Syntactic dependency	MT classifier	91.5	91.8	89.6	93.4	94.4
	Majority	69.0	68.6	59.4	67.1	72.4
	OOV rate	10.3	12.9	21.7	5.9	10.9
CCG tags	Xu, Auli, and Clark (2015)	–	–	–	93.1	–
	MT classifier	–	–	–	91.9	–
	Majority	–	–	–	72.1	–
	OOV rate	–	–	–	6.9	–

we directly pit the representations trained on different translation units against each other and compare their performance in predicting syntactic properties. Figure 8 shows that representations learned from subword units (BPE and Morfessor) consistently outperform the ones learned from character units in both tasks (CCG and syntactic dependency labeling), reinforcing the results found by Sennrich (2017). Character-based models, on the other hand, do better than word-based models, which could be attributed to unknown words (in the word-based models). We found subword units, particularly those obtained using a morpheme-based segmentation, to give the best results. This could be because the linguistically motivated subword units are more aligned with the syntactic task than the compression-based BPE segmentation.

A possible confound is that character-based models start from a lower linguistic level compared to word or subword models and may require more depth to learn long-range dependencies. To verify this, we trained 3-layered character models for Czech-to-English and English-to-German. We extracted feature representations and trained

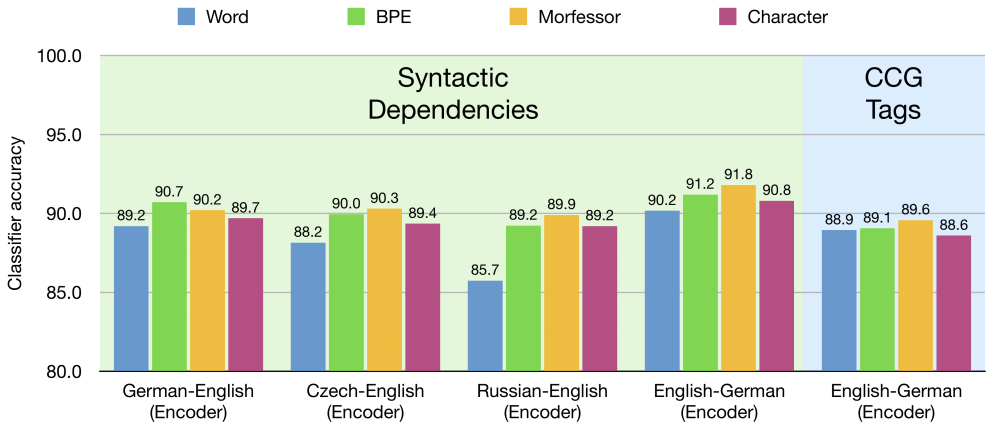
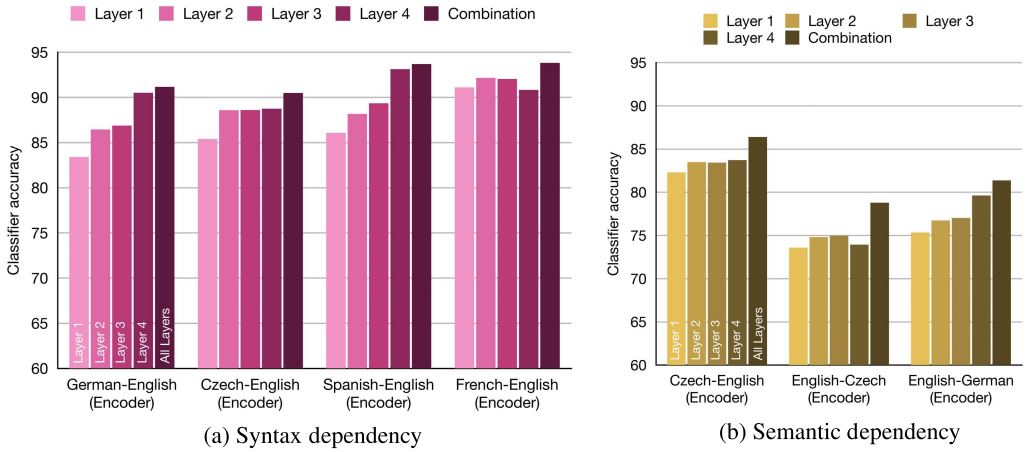


Figure 8
Dependency labeling and CCG supertagging accuracy using encoder representations obtained from NMT models trained with different translation units on the source side; the target side is fixed as BPE. Subword units generate better representations than character- or word-based ones.



(a) Syntax dependency (b) Semantic dependency

Figure 9
Syntactic and semantic dependency labeling accuracy using representations from layers 1 to 4, taken from encoders of BPE-based NMT models in different language pairs. Higher layers generate better representations for these tasks in most cases, but a combination of all layers works best, indicating that some relevant information is also captured in the lower layers.

classifiers to predict syntactic dependency labels. Our results show that using an additional layer does improve the prediction accuracy, giving the same result as subword segmentation (Morfessor) in the case of Czech-to-English, but still worse in the case of English-to-Czech (see Table A.4 in the Appendix for results).

7.2 Effect of Network Depth

We previously found that morphology is predominantly being captured in layer 1 of the NMT models. We now repeat the experiments for syntactic dependencies. Figure 9a shows the results of predicting syntactic dependency labels using representations from different layers in the trained models. We found that representations from layer 4 performed better than representations from lower layers except for the French encoder, where layer 3 performs better. We also repeated this experiment with CCG supertags (see Table A.5 in the supplementary material) and found that higher layers (3 and 4) consistently outperform lower layers and except for English-Czech, the final layer gives the best accuracy in all cases.[16] These results are consistent with the syntactic dependency results. We repeated these experiments with the multi-parallel UN corpus by training English-to-{French, Arabic, Spanish, Russian, and English} bilingual models. Comparing successive layers (for example, comparing layer 2 versus layer 3), in the majority of the cases, the higher layer performed statistically significantly better than the lower one ($\rho < 0.01$), according to the approximate randomization test (Padó 2006).[17] Similar to the results on morphological tagging, a combination of all layers achieved the best results. See the **Combination** bar in Figure 9a. This implies that although syntax is

16 In their study of NMT and language model representations, Zhang and Bowman (2018) noticed that POS is better represented at layer 1 whereas CCG supertags are sometimes, but not always, better represented at layer 2 (out of 2-layer encoders).

17 See Section 11 in the supplementary information for the detailed results.

mainly learned at higher layers, syntactic information is at least partly distributed across the network.

One possible concern with these results is that they may be appearing because of the stacked RNN layers, and not necessarily due to the translation task. In the extreme case, perhaps even a random computation that is performed in stacked RNN layers would lead to improved performance in higher layers. This may be especially concerning when predicting relation labels, as this requires combining information about two words in the sentence. To verify that the actual translation task is important, we can look at the performance with random models, initialized in the same manner but not trained at all. Figure 10 shows that higher layers in random networks generally generate worse representations. Layer 1 does improve the performance compared to layer 0 (word embeddings without contextual information) showing that some information is captured even in random models. However, after layer 1 the performance degrades drastically, demonstrating that higher layers in random models do not generate informative representations.

The experiment with random weights shows that training the NMT system is important. Does the actual translation task matter? Figure 10 also shows the results using representations from English-to-English models, that is, an autoencoder scenario. There

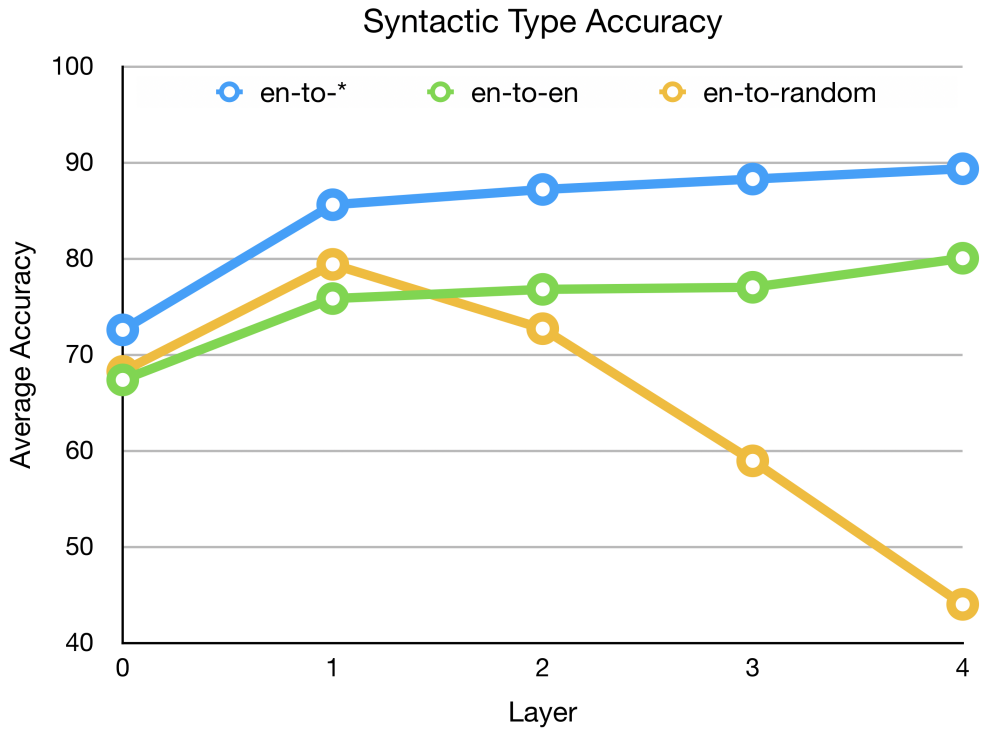


Figure 10
Syntactic dependency labeling results with representations from different encoding layers of (word-based) NMT models trained on translating English to other languages (en-to-*, averaged over target languages), compared with an auto-encoder (en-to-en) and to untrained modeled with random weights (en-to-* rand). The MT-trained representations improve with each layer, while the random representations degrade after layer 1. The auto-encoder representations also improve but are below the MT-trained ones. These results show that learning to *translate* is important for obtaining good representations.

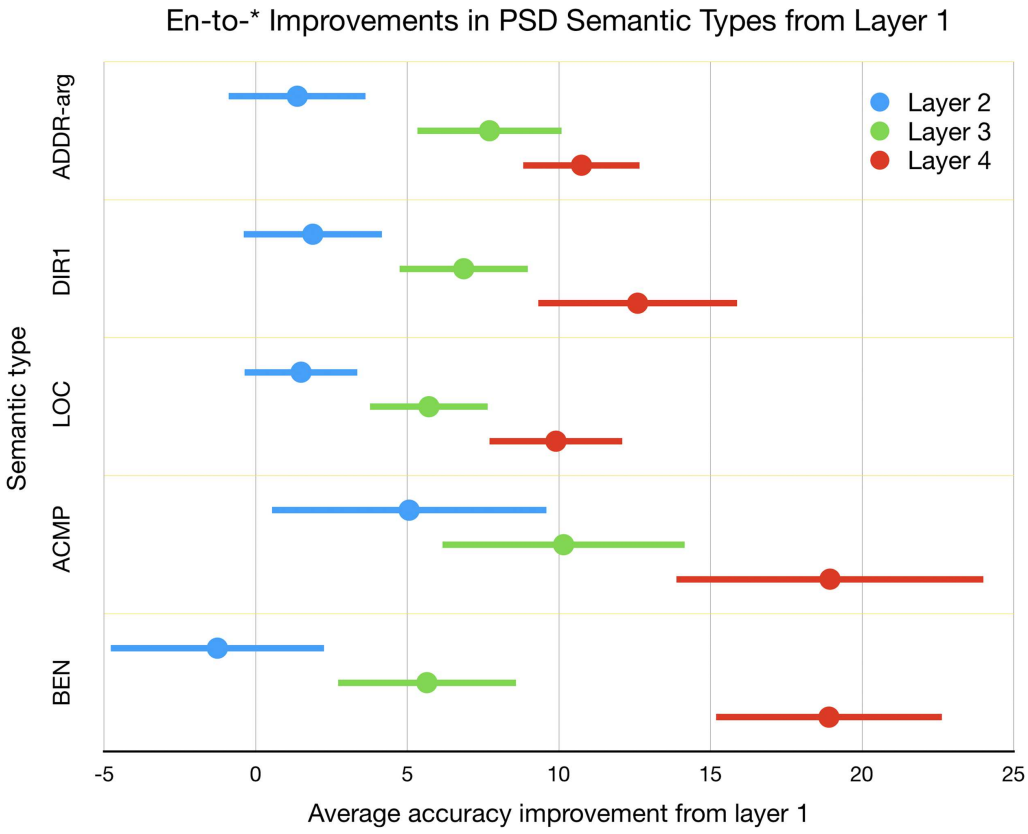


Figure 11
Improvement in accuracy of syntactic relation labeling with layers 2/3/4 compared with layer 1. The figure shows the five most improved relations when averaging results obtained with encoder representations from (word-based) NMT models trained on translating English to other languages.

is a notable degradation in representation quality when comparing the autoencoder results to those of the machine translation models. For example, the best results for predicting syntactic dependencies with the autoencoder are around 80% at layer 4. In contrast, the same layer in the translation models produces a score of 88%. In general, the representations from the machine translation models are always better than those from the autoencoder, and this gap increases as we go higher in the layers. This trend is similar to the results on morphological and semantic tagging with representations from autoencoders that were reported previously.

7.3 Analysis

In this section, we analyze two aspects of how information on syntactic dependencies is captured in different NMT layers: how different types of relations are represented and what the effect of head-modifier distance is. The results in this section are obtained using models trained on the United Nations corpus, as described in Section 5.1.

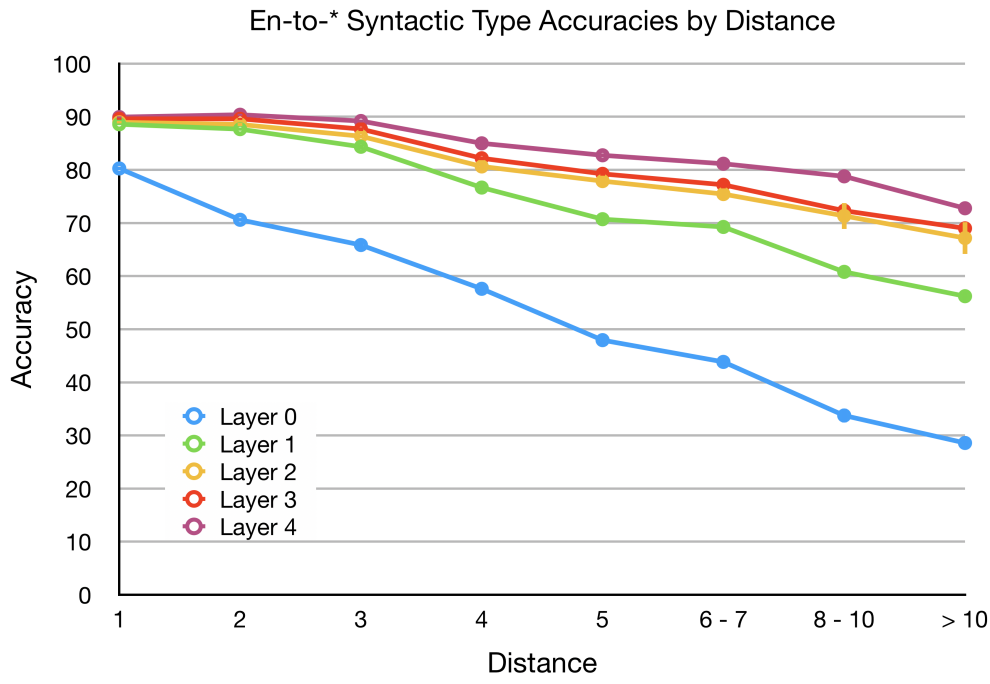


Figure 12
Results of syntactic relation labeling at different relation distances using encoder representations from different layers of (word-based) NMT models trained on translating English to other languages (averaged over target language). Long-distance relations are difficult to capture at all layers, but higher layers mitigate more of the drop in performance.

7.3.1 Effect of Relation Type. When are higher-layer representations especially important for syntactic relations? Figure 11 breaks down the performance according to the type of syntactic relations. The figure shows the five relations that benefit most from higher layer representations.[18] The general trend is that the quality of the representation improves with higher layers, with up to 20−25% improvement with representations from layer 4 compared with layer 1. The improvement is larger for certain relations: dependent clauses (`advcl, ccomp`), loose relations (`list, parataxis`), and other typically long-range dependencies such as conjunctions (`conj`) and appositions (`appos`). Core nominal arguments like subject (`nsubj`) and object (`obj`) also show consistent improvements with higher layers. Relations that do not benefit much from higher layers are mostly function words (`aux, cop, det`), which are local relations by nature, and the relation between a conjunct and the conjunction (`cc`), as opposed to the relation between two conjuncts (`conj`). These relations are local by nature and also typically less ambiguous. For example, the relation between a conjunction `and` and a noun is always labeled as `cc`, whereas a verb and a noun may have a subject or object relation.

7.3.2 Effect of Relation Distance. In order to quantify the notions of global and local relations, let us consider relation distance. Figure 12 shows the representation quality

18 The results shown are with English dependencies using NMT models trained on English to other languages, but the trends are similar for other language pairs.

as a function of the distance between the words participating in the relation. Predicting long-distance relations is clearly more difficult than predicting short-distance ones. As the distance between the words in the relation grows, the quality of the representations decreases. When no context is available (layer 0, corresponding to word embeddings), the performance quickly drops with longer distance relations. The drop is more moderate in the hidden layers, but in low layers the effect of relation distance can still be as high as 25%. Higher layers of the network mitigate this effect and bring the decrease down to under 5%. Moreover, every layer is performing better than the previous one at each distance group. This indicates that higher layers are much better at capturing long-distance syntactic information.

8. Semantics Results

We now study how information on meaning is captured in NMT models in the context of lexical semantic (SEM) tagging and semantic dependency labeling tasks (refer to Section 3.3 for details on the tasks). We study the following specific questions: (i) Do NMT systems learn informative semantic representations? (ii) Can a neural network model learn to map a sequence of subwords or character symbols to a meaning representation? and (iii) What layers in the model learn more about semantic tags and relations?

The experiments reported in this section on are conducted mainly on English, as the semantic tagging task and data set are recent developments that were initially only available in English. We also experiment and report results for German, for which a new semantic tagging data set is being developed. However, as the German annotations are very sparse (see Section 5.4), we performed a cross-fold evaluation when reporting results for German. For the semantic dependency labeling, we additionally used Czech data to strengthen the empirical evidence.

In this section we only report *encoder-side* representation as the the analysis of *decoder-side* representations requires parallel data to generate the hidden representations and no standard tools for annotating the data exist. Table 10 shows the results. The classifier achieves 91.4% on the semantic tagging task and 85% and 80% on the task of semantic labeling for Czech and English, respectively. All results are significantly better than the local majority baseline, suggesting that NMT representations learn substantial semantic information.

Table 10
Local majority baseline (most frequent tag/label) and classification accuracy using encoder representations generated by the NMT models, on semantic tasks. The models are trained on translating Czech to English (cs column) or English to German (en column). The classifier results are far superior to the majority baseline, indicating that NMT representations contain a non-trivial amount of semantic information.

		cs	en
Semantic dependencies	MT classifier	87.8	81.5
	Majority	63.1	57.3
	OOV rate	12.1	6.3
Semantic tags	Bjerva, Plank, and Bos (2016)	–	95.2
	MT classifier	–	93.4
	Majority	–	84.2
	OOV rate	–	4.1

8.1 Impact of Translation Unit on Learning Semantics

Next we investigate whether the representations learned from characters or subword units can effectively model semantic information. We trained classifiers using the representations generated from different NMT models that were trained using character or subword units (BPE and Morfessor). Figure 13 summarizes the results on the semantic dependency labeling task and the semantic tagging task. In the semantic dependency labeling task, the character-based models perform significantly worse compared with the word-based and subword-based counterparts. We found that using subword-based representations, particularly morpheme-based segmentation, gives better performance in most scenarios. These results are in contrast with morphological tagging results, where character-based representations were consistently and significantly better compared with their subword counterparts. On comparing the prediction results between subword and character-based representations, we found that in many cases, character-based models failed to predict the label correctly when the head and modifier words are further apart, that is, in the case of long-distance dependencies. However, this was not always true as in some cases character-based models were able to correctly predict the dependency label for a head that was 12 words apart.

On semantic tags, subword-based (BPE and Morfessor) representations and character-based representation achieve comparable results for English. However, for German, BPE-based representations performed better than the other representations.

8.2 Effect of Network Depth

We found the representations learned in the lower encoding layer to perform better on the task of morphological tagging. Here we investigate the quality of representations at different encoding layers, from the perspective of semantic properties.

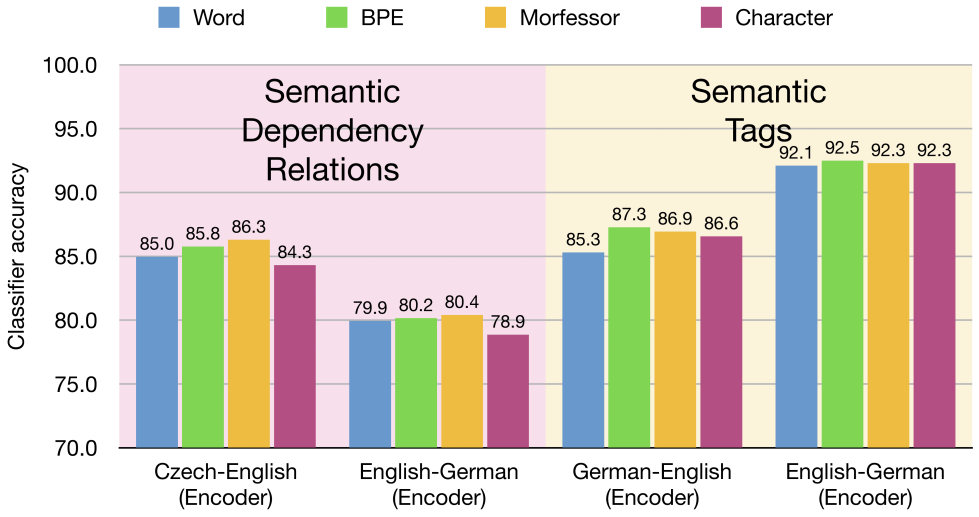


Figure 13
Semantic tagging and dependency labeling results using representations of NMT models trained with different translation units on the source side; the target side is always BPE.

Table 11
Semantic tagging accuracy using features from the k-th encoding layer of 4-layered NMT models trained with different target languages. "En" column is an English autoencoder. BLEU scores are given for reference. Statistically significant differences from layer 1 are shown at $p < 0.001^{(*)}$ and $p < 0.01^{(**)}$. See text for details.

k	Ar	Es	Fr	Ru	En
		SEM tagging accuracy			
0	81.9*	81.9*	81.8*	81.8*	81.2*
1	87.9	87.7	87.8	87.9	84.5
2	87.4*	87.5*	87.4*	87.3*	83.2*
3	87.8	87.9*	87.9**	87.3*	82.9*
4	88.3*	88.6*	88.4*	88.1*	82.1*
			BLEU		
	32.7	49.1	38.5	34.2	96.6

Concerning lexical semantic tagging, as Table 11 shows, representations from layers 2 and 3 do not consistently improve performance above layer 1. However, representations from layer 4 lead to small but significant improvement with all target languages, according to the approximate randomization test.[19] We observed a similar pattern in the case of semantic dependency labeling tasks (see Figure 9b), where higher layers (layer 4 in the case of Czech-English and English-German and layer 3 in the case of English-Czech) gave better accuracy. Intuitively, higher layers have a more global perspective because they have access to higher representations of the word and its context, while lower layers have a more local perspective. Layer 1 has access to context but only through one hidden layer, which may not be sufficient for capturing semantics. It appears that higher representations are necessary for learning even relatively simple lexical semantics and, especially, predicate–argument relations.

8.3 Analysis of Lexical Semantics

In this section, we analyze three aspects of lexical semantic information as represented in the semantic tagging data set. First, we categorize semantic tags into coarse-grained categories and compare the classification quality within and across categories. Second, we perform a qualitative analysis of discourse relations and when they are better represented in different NMT layers. Third, we compare the quality of encoder representations when translating into different target languages. The results in this section are obtained using models trained on the United Nations corpus, as described in Section 5.1.

8.3.1 Semantic Tag Level Analysis. The SEM tags are grouped in coarse-grained categories such as events, names, time, and logical expressions. Figure 14 shows the change in F_1 score (averaged over target languages) when moving from layer 1 to layer 4 representations. The blue bars describe the differences per coarse tag when directly predicting coarse tags. The red bars show the same differences when predicting fine-grained tags and micro-averaging inside each coarse tag. The former shows the differences between

19 These results are obtained using models trained on the United Nations multi-parallel corpus.

the two layers at distinguishing among coarse tags. The latter gives an idea of the differences when distinguishing between fine-grained tags within a coarse category. The first observation is that in the majority of cases there is an advantage for classifiers trained with layer 4 representations, that is, higher layer representations are better suited for learning the SEM tags, at both coarse and fine-grained levels.

Considering specific tags, higher layers of the NMT model are especially better at capturing semantic information such as *discourse relations* (DIS tag: subordinate vs. coordinate vs. apposition relations), semantic properties of nouns (*roles* vs. *concepts*, within the ENT tag), *events* and *predicate tense* (EVE and TNS tags), *logic relations* and *quantifiers* (LOG tag: disjunction, conjunction, implication, existential, universal, etc.), and *comparative constructions* (COM tag: equatives, comparatives, and superlatives). These examples represent semantic concepts and relations that require a level of abstraction going beyond the lexeme or word form, and thus might be better represented in higher layers in the deep network.

8.3.2 Analyzing Discourse Relations. Now we analyze specific cases of disagreement between predictions using representations from layer 1 and layer 4. We focus on discourse relations, as they show the largest improvement when going from layer 1 to layer 4 representations (DIS category in Figure 14). Intuitively, identifying discourse relations requires a relatively large context so it is expected that higher layers would perform better in this case.

There are three discourse relations in the SEM tags annotation scheme: subordinate (SUB), coordinate (COO), and apposition (APP) relations. For each of those, Figure 15

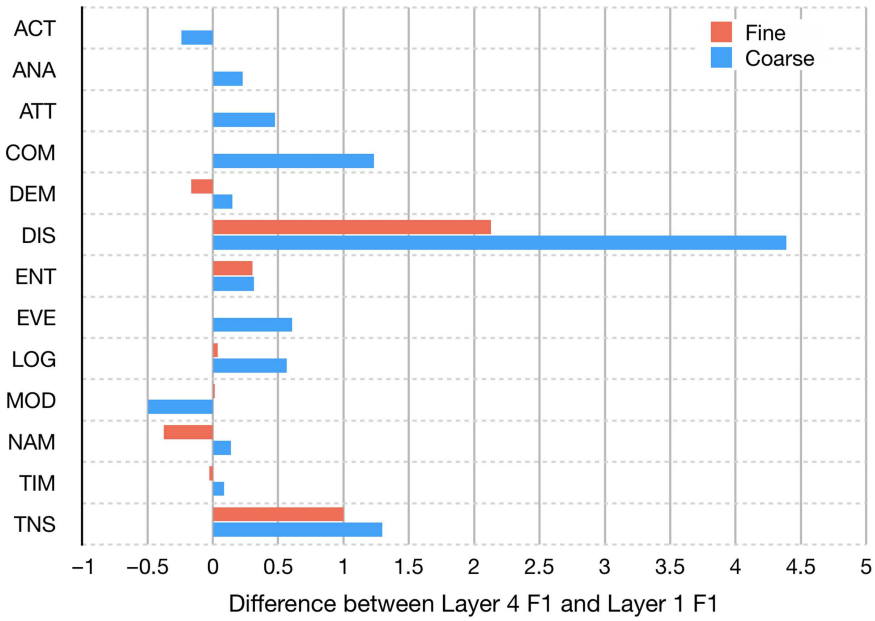


Figure 14
Difference in semantic tagging F_1 when using representations from layer 4 compared to layer 1, showing F_1 when directly predicting coarse tags (blue) and when predicting fine-grained tags and averaging inside each coarse tag (red). The representations are taken from encoders of (word-based) NMT models trained on translating English to other languages (averaged over target language).

(examples 1–9) shows the first three cases in the test set where layer 4 representations correctly predicted the tag but layer 1 representations were wrong. Examples 1–3 have subordinate conjunctions (*as*, *after*, *because*) connecting a main and an embedded clause, which layer 4 is able to correctly predict. Layer 1 mistakes these as attribute tags (REL, IST) that are usually used for prepositions. In examples 4–5, the coordinate conjunction *and* is used to connect sentences/clauses, which layer 4 correctly tags as COO. Layer 1 wrongly predicts the tag AND, which is used for conjunctions connecting shorter expressions like words (e.g., "murder *and* sabotage" in example 1). Example 6 is probably an annotation error, as *and* connects the phrases "lame gait" and "wrinkled skin" and should be tagged as AND. In this case, layer 1 is actually correct. In examples 7–9, layer 4 correctly identifies the comma as introducing an apposition, while layer 1 predicts NIL, a tag for punctuation marks without semantic content (e.g., end-of-sentence period). As expected, in most of these cases identifying the discourse function requires a fairly large context.

Finally, we show in examples 10–12 the first three occurrences of AND in the test set, where layer 1 was correct and layer 4 was wrong. Interestingly, two of these (10, 11) are clear cases of *and* connecting clauses or sentences, which should have been annotated as COO, and the last (12) is a conjunction of two gerunds. The predictions from layer 4 in these cases thus appear justifiable.

8.3.3 Effect of Target Language. Does translating into different languages make the NMT system learn different source-side representations? We previously found a fairly consistent effect of the target language on the quality of encoder representations for POS and morphological tagging, with differences of ∼2–3% in accuracy. Here we examine whether such an effect exists in SEM tagging. We trained 4-layered English-to-{Arabic,

	L1	L4	
1	REL	*SUB*	Zimbabwe 's President Robert Mugabe has freed three men who were jailed for murder and sabotage *as* they battled South Africa 's anti-apartheid African National Congress in 1988 .
2	REL	*SUB*	The military says the battle erupted *after* gunmen fired on U.S. troops and Afghan police investigating a reported beating of a villager .
3	IST	*SUB*	Election authorities had previously told Haitian-born Dumarsais Simeus that he was not eligible to run *because* he holds U.S. citizenship .
4	AND	*COO*	Fifty people representing 26 countries took the Oath of Allegiance this week (Thursday) *and* became U.S. citizens in a special ceremony at the Newseum in Washington , D.C.
5	AND	*COO*	But rebel groups said on Sunday they would not sign *and* insisted on changes .
6	AND	*COO*	A Fox asked him , " How can you pretend to prescribe for others , when you are unable to heal your own lame gait *and* wrinkled skin ? "
7	NIL	*APP*	But Syria 's president , Bashar al-Assad , has already rejected the commission 's request [...]
8	NIL	*APP*	Hassan Halemi , head of the pathology department at Kabul University where the autopsies were carried out , said hours of testing Saturday confirmed [...]
9	NIL	*APP*	Mr. Hu made the comments Tuesday during a meeting with Ichiro Ozawa , the leader of Japan 's main opposition party .
10	*AND*	COO	[...] abortion opponents will march past the U.S. Capitol *and* end outside the Supreme Court .
11	*AND*	COO	Van Schalkwyk said no new coal-fired power stations would be approved unless they use technology that captures *and* stores carbon emissions .
12	*AND*	COO	A MEMBER of the Kansas Legislature meeting a Cake of Soap was passing it by without recognition , but the Cake of Soap insisted on stopping *and* shaking hands .

Figure 15
Examples of cases of disagreement between layer 1 (L1) and layer 4 (L4) representations when predicting semantic tags. The correct tag is *italicized* and the relevant word is *underlined*.

Russian, French, Spanish, English} models with the multi-parallel UN corpus. Table 11 shows results using features obtained by training NMT systems on different target languages (the English source remains fixed). There are very small differences with different target languages (∼0.5%). While the differences are small, they are mostly statistically significant. For example, at layer 4, all the pairwise comparisons with different target languages are statistically significant ($p < 0.001$). Again, we note that training an English autoencoder results in much worse representations compared with MT models. In contrast, the autoencoder has excellent sentence recreation capability (96.6 BLEU). This indicates that learning to translate (to any foreign language) is important for obtaining useful representations for semantic tagging, as it is for morphological tagging.

8.4 Analysis of Semantic Dependencies

In the previous sections, we analyzed how the layer depth impacts the representations from the perspective of specific syntactic relations (Section 7.3.1) and lexical semantic tags (Section 8.3.1). We found that higher layers tend to better represent properties that are more global, loose, and abstract compared with lower layers. Does the same hold for semantic dependencies?

Figure 16 shows the improvement from higher layer representations. The five most improved relations are highlighted. The following are examples from the PSD manual (Cinková et al. 2004). We give in parentheses the average distance in words between head and modifier, per relation. Four of the top five relations are looser kinds of relations that syntactically correspond to adjuncts: accompaniment (ACMP; "He works without his glasses"; distance 6.93); to whose advantage something happens (BEN; "He did it for their sake"; distance 4.40); direction (DIR1; "He made a step from the wall"; distance 4.41); and location (LOC; "a match in a foreign country"; distance 4.77); The fifth is an addressee argument (ADDR-arg; "He gave the child a toy."; distance 3.69). These relations also have longer distances between head and modifier compared to the overall average distance (3.34 words).

In contrast, the relations that benefit the least (highlighted at the top of Figure 16) include a disjunctive that captures the relation between the disjunction "or" and a word in a list (DISJ.member; distance 2.22); expressing difference (DIFF; "The goods were delivered four days later"; distance 2.54); a degree specifier for expressing extent (EXT; e.g., the relation between "about" or "almost" and a quantity; distance 1.87); a rhematizer that often connects a negation word to its negated verb (RHEM; "Cray Research did not want to...", example from the PSD data set; distance 2.17); and linking the clause to the preceding text (PREC; "Hence, I'm happy"; distance 7.73). Of these, the first three actually drop in performance when using (some of the) higher layer representations, which may be explained by their more local, tight relation. This also partly accords with the distances, which are below average for the bottom 4. The PREC relation is an exception, with a relatively large distance but almost no benefit from higher layers. This can be explained by its use for words linking the clause to the preceding text, which are limited and easy to memorize ("However", "Nevertheless", "Moreover", etc.). As these cases are assigned the main verb as the PREC head, they may span over large distances, but their closed-class nature enables simple memorization even at low layers. However, relation distance does not explain the entire benefit from higher layers, as some of the most distant relations are not the ones that benefit most from higher layers. Still, similar to the syntactic dependencies, the representations from higher layers benefit more in case of looser, less tight semantic relations.

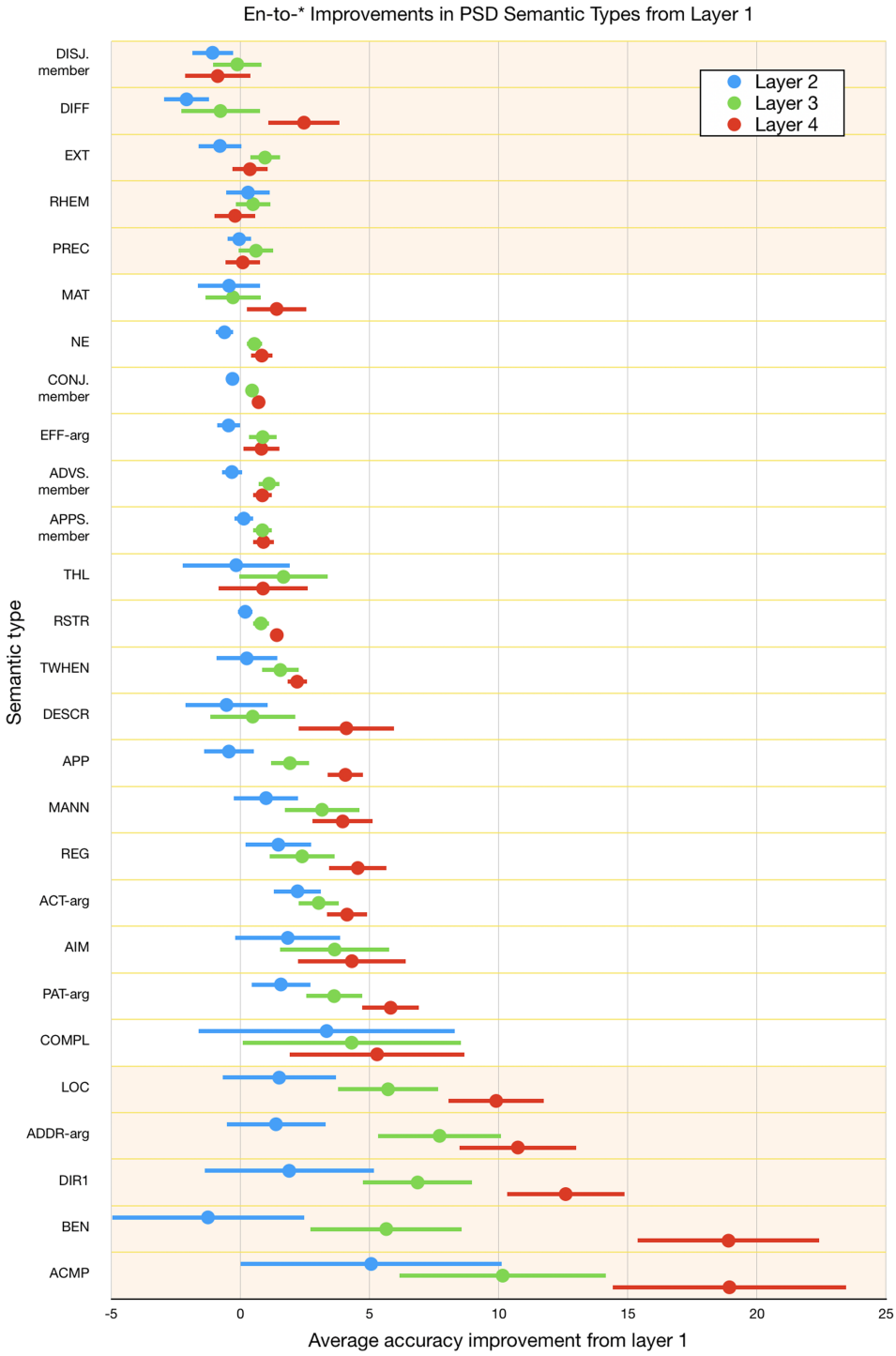


Figure 16
Improvement in accuracy of semantic relation labeling with layers 2/3/4 compared with layer 1, when averaging results obtained with encoder representations from (word-based) NMT models trained on translating English to other languages. The five least/most improved relations are highlighted.

9. Comparison Against Multilingual Models

Languages share orthography, morphological patterns, and grammar. Learning to translate between several pairs simultaneously can help improve the translation quality of the underlying languages and also enable translation for language pairs with little or no parallel data. Johnson et al. (2017) exploited a remarkably simple idea of appending a language code to each training sentence, in a shared encoder-decoder framework. The models are trained with all multilingual data consisting of multiple language pairs at once. In their projection of a small corpus of 74 triples of semantically identical cross-language (Japanese, Korean, English) phrases to 3D space via t-SNE, they found preliminary evidence that the shared architecture in their multilingual NMT systems learns a universal interlingua. We use this idea with our machinery to investigate how effective the multilingual representations are in learning morphology, syntax, and semantics, compared to their corresponding bilingual models.

We trained 4-layer multilingual systems with many-to-one and one-to-many configurations with English on one side (encoder/decoder) and German, Spanish, French, and Czech on the other side. The models were trained with BPE subword units. We trained two versions of the multilingual model, one with the same parameters as the bilingual models and another with word embeddings and LSTM states with 1,024 dimensions (as opposed to the default 500 dimensions). Our goal was to investigate the effect of increasing model parameters on translation quality and representation quality in terms of the understudied linguistic phenomenon. The systems were trained on NEWS+TED+Europarl, approximately 2.5M sentences per language pair, and a total of 10M sentences for training the multilingual models.

Figure 17 shows BLEU scores comparing bilingual and multilingual translation systems across different language pairs. We see that the many-to-one multilingual system (i.e., *-to-English) is on par or slightly behind bilingual systems when trained with the same number of parameters as the bilingual models. In contrast, the one-to-many multilingual system (i.e., English-to-*) is significantly worse compared with its bilingual counterparts. The reason for this discrepancy could be that generation is a harder task than encoding, especially when translating into morphologically-rich languages: An average difference of -1.35 is observed when translating out of English compared with -0.13 when translating into English. The larger multilingual models (with twice as many parameters) restored the baseline performance, in fact showing significant improvements in many cases. We also trained two of the bilingual baselines (Czech $\leftrightarrow$ English) by doubling the parameters. While the large multilingual system gave an improvement of +1.4 (see Table 12) over the baseline multilingual system by

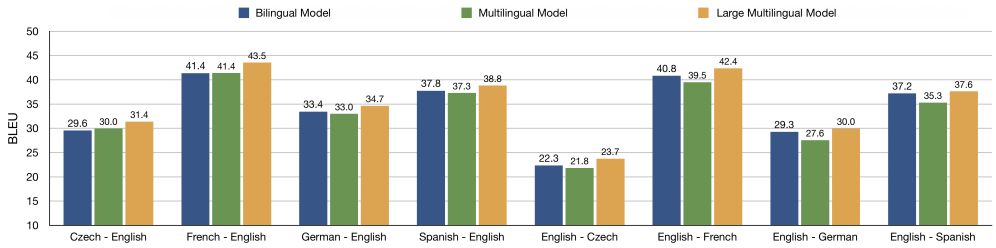


Figure 17
Comparing bilingual and multilingual systems in terms of translation quality. The multilingual models are many-to-one (*-to-English) or one-to-many (English-to-*) during training, but run on a specific language pair during testing.

Table 12
Comparing bilingual and multilingual Czech↔English models across translation quality and classifying different linguistic properties. bi = bilingual model; bi* = larger bilingual model; multi = multilingual model; multi* = larger multilingual model. Larger models have twice the number of word embedding and LSTM state dimensions. The larger models improve substantially the multilingual systems, but only slightly improve the bilingual systems. Morphology = morphological tagging; Syntax = syntactic dependency labeling; Semantics = semantic dependency labeling.

	cs-en				en-cs			
	BLEU	Morphology	Syntax	Semantics	BLEU	Morphology	Syntax	Semantics
bi	29.6	78.9	90.6	86.4	22.3	81.8	–	78.8
bi*	29.8	77.7	91.2	84.6	22.6	82.5	–	79.3
multi	30.0	82.1	91.8	86.8	21.8	81.8	–	81.2
multi*	31.4	84.0	88.7	87.8	23.7	84.6	–	81.4

doubling the parameters size, the bilingual system only obtained an improvement of +0.2 by increasing the model size. A similar pattern was observed in the opposite direction where increasing the model size gave a BLEU improvement of +1.9 in the multilingual system, but only +0.3 in the case of the bilingual system. These results show that multilingual systems benefit from other language pairs being trained in tandem and suggest that the underlying representations are richer than the ones trained using bilingual models. We now proceed to analyze these representations in light of the understudied linguistic properties.

Figures 18, 19a, and 19b show that the representations learned from multilingual models, despite sharing the encoder and decoder representations among four languages, can still effectively learn the same amount of morphology, syntax, and semantics as learned by their bilingual counterparts. In almost all cases, the multilingual representations are either better or at par with the bilingual models. Using a larger multilingual model (double the parameters size) gave consistent improvements in accuracy, which resonate with the improvement in translation quality.

Focusing on morphology (Figure 18), a multilingual encoder generates representations that are better than its bilingual counterpart in most cases, while a multilingual decoder slightly degrades representation quality compared to the bilingual decoder. However, using a larger multilingual model leads to substantial improvements. These results mirror the patterns shown in terms of translation quality (Figure 17). In the case of syntactic and semantic dependencies (Figures 19a and 19b), even the default-sized multilingual model works better than the bilingual model, and the larger multilingual

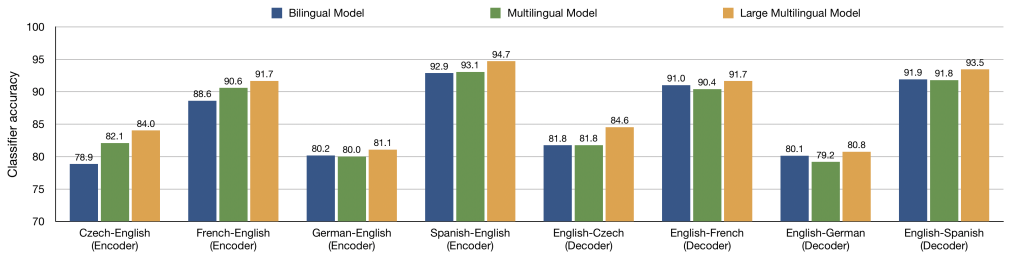
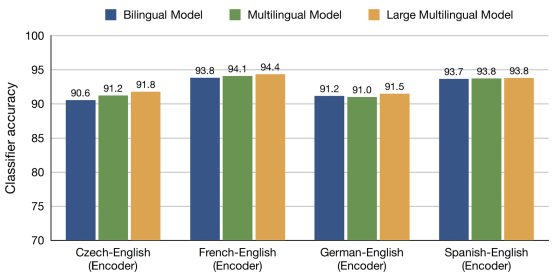
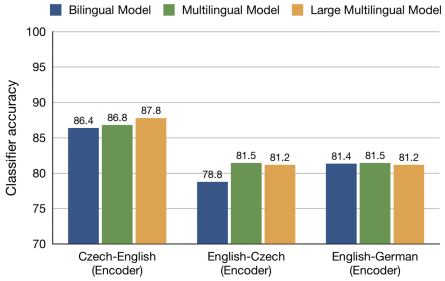


Figure 18
Comparing bilingual and multilingual models in terms of morphological tagging accuracy. Same-size multilingual models benefit the encoder representations compared to bilingual models, but not decoder representations; larger multilingual models benefit also the decoder.



(a) Syntactic dependency labeling.



(b) Semantic dependency labeling.

Figure 19
Comparing bilingual and multilingual models with syntactic and semantic dependencies. Multilingual models consistently improve the NMT representations.

model leads to additional small improvements in syntactic dependencies and in Czech semantic dependencies (the Czech-English bars in Figure 19b). The larger multilingual model does not improve in the case of English semantic dependencies (English-Czech/German bars in Figure 19b), even though the corresponding BLEU scores do improve with a larger multilingual model (Figure 17).[20]

In an effort to probe whether increasing the number of parameters in a bilingual model would result in similar performance improvements, Table 12 shows results across different properties in Czech↔English language pairs. We consistently found that increasing the model size does not lead to the same improvements as we observed in the case of multilingual models. These results reinforce that multilingual NMT models learn richer representations compared to the bilingual model and benefit from the shared properties across different language pairs.

10. Discussion

In this section, we discuss the overall patterns that arise from the experimental results from several angles. First, we discuss how to assess the overall quality of the learned NMT representations with regard to other baselines and upper bounds. Second, we consider NMT representations from the perspective of contextualized word representations and contrast them to recent popular representations. Third, we reflect on the methodological approach taken in this work, and what it may or may not tell us about how the NMT model exploits language representations. Fourth, we briefly discuss the relation of our results to other NMT architectures. Finally, we touch upon the role of analysis work in understanding and improving NMT models.

10.1 Assessing Representation Quality

The analyses presented in this work shed light on the quality of different language representations in NMT, with a particular focus on comparing various NMT components and design choices (layers, translation units, etc.). Our questions, therefore, have mostly

20 One speculation for this might be that translating into several target languages does not add much semantic information on the source side because this kind of information is more language-agnostic, but at this point there is insufficient evidence for this kind of claim.

been from a comparative perspective: How good are certain NMT representations compared with others with respect to certain linguistic tasks. But just how good are the NMT representations overall? Answers to this question may depend on the use case. One could, for example, evaluate the utility of NMT representations to improve state-of-the-art performance by plugging them as additional features in some strong model. Indeed, McCann et al. (2017) found this approach to yield state-of-the-art results in several language understanding tasks. It is also important to consider the quality of the NMT representations for the understudied tasks here in comparison to other baselines and competitive systems. Throughout the article, we have compared the results to a majority baseline, arguing that NMT representations obtain substantial improvements. Here we compare, for each linguistic task, the best performing NMT representations with several baselines and upper bounds. We compare with the local majority baseline (most frequent tag/label for each word according to the training data, and the globally most frequent tag/label for words not seen in training) and with a classifier trained on word embeddings that are pre-trained on the source-side of the MT training data. We also train an encoder-decoder on converting text to tags, by automatically annotating the source side of the MT parallel data. Then we use this encoder-decoder to tag the test set of the supervised data and evaluate its quality. Finally, we generate representations from the encoder of this encoder-decoder model and train a classifier on them to predict the tags. This setting aims to mimic our main scenario, except that we generate representations with an encoder-decoder specially trained on the linguistic task that we evaluate, rather than representations generated by an NMT model.

Table 5 shows the results. A classifier trained on NMT representations performs far better than the majority baseline, as we have already confirmed. A similar classifier trained on representations from a task-specific encoder-decoder performs even better. This indicates that training on a specific task leads to representations more geared toward that task, as may be expected. In fact, a similar behavior has been noted with other contextual word representations (Liu et al. 2019). Still, the representations do not contain all available information (or, not all information may be extracted by a simple classifier), as the task-specific encoder-decoder performs better than a classifier trained on its representations.

10.2 Contextualized Word Representations

The representations generated by NMT models may be thought of as contextualized word representations (CWRs), as they capture context via the NMT encoder or decoder. We have already mentioned one work exploiting this idea, known as CoVE (McCann et al. 2017), which used NMT representations as features in other models to perform various NLP tasks. Other prominent contextualizers include ELMo (Peters et al. 2018a), which trains two separate, forward and backward LSTM language models (with a character CNN building block) and concatenates their representations across several layers; GPT (Radford et al. 2018) and GPT-2 (Radford et al. 2019), which use transformer language models based on self-attention (Vaswani et al. 2017); and BERT (Devlin et al. 2019), which uses a bidirectional transformer model trained on masked language modeling (filling the blanks). All these generate representations that feed into task-specific classifiers, potentially with fine-tuning the contextualizer weights.[21]

21 See Peters, Ruder, and Smith (2019) for an evaluation of when it is worthwhile to fine-tune.

How do NMT representations compare with CWRs trained from raw text? Directly answering this question is beyond the scope of this work, and is also tricky to perform for two reasons. First, CWRs like ELMo, BERT, and GPT require very large amounts of data, on the order of billions of words, which is far beyond what typical NMT systems are trained on (our largest NMT systems are trained on an order of magnitude fewer words). Second, at present, various CWRs are trained in incomparable settings in terms of data, number of parameters, and infrastructure. It seems that a comparison of common CWRs themselves is necessary before they are compared to NMT representations.

There is, however, indirect information that may tell us something about how CWRs trained on raw texts behave in comparison to NMT representations. Bowman et al. (2018) compared sentence encoders trained on a variety of tasks, including the CoVE translation representations, and evaluated on language understanding tasks. They found language modeling pre-training to perform best, but cautioned that without fine-tuning, many of the results are not far above trivial baselines. They also found that grammar-related tasks benefit more from such pre-training than meaning-oriented tasks. Peters et al. (2018b) compared the ELMo LSTM with similar systems based on convolutions or transformer-style self-attention. They found that all architectures learn hierarchical representations: The word embedding layer focuses on morphology, low encoding layers focus on local syntax, and high encoding layers carry more semantic information. These results are mostly in line with our findings concerning representation depth, although we have not noticed a clear separation between syntactic and semantic properties. Zhang and Bowman (2018) compared representations from NMT and bidirectional language models on POS tagging and CCG supertagging. They found the language model representations to consistently outperform those from NMT. In other work, we have found that language model representations are of similar quality to NMT ones in terms of POS and morphology, but are behind in terms of semantic tagging (Dalvi et al. 2019a).

Tenney et al. (2019) compared representations from CoVE, ELMo, GPT, and BERT on a number of classiciation tasks, partially overlapping with the ones we study. They found that CWRs trained on raw texts outperform the MT representations of CoVE; however, as noted above these models are all trained in very different setups and cannot be fairly compared. Another interesting finding is that learning a weighted mix of layers works better than any one layer, and also better than concatenating. This again indicates that some layers are better than others for different tasks, consistent with our results. Concerning different tasks, Tenney et al. (2019) found that CWRs are especially helpful (compared with a lexical baseline) with syntactic tasks, such as dependency and constituent labeling, and less helpful with certain semantic tasks like capturing fine-grained semantic attributes and pronoun resolution. They did notice improvements with semantic roles, which are related to our predicate–argument relations, where we also noticed significant improvements at higher layers. Finally, Liu et al. (2019) compared ELMo, GPT, and BERT on various classification tasks in terms of their linguistic knowledge and transferability. They found that a simple classifier trained on top of the (frozen) representations led to state-of-the-art results in many cases, but failed on tasks requiring fine-grained linguistic knowledge like conjunct identification. They observed that the first layer of LSTM-based CWRs performs better than other layers, while in transformer-based models the intermediate layers are the best. Considering different pre-training tasks, higher layer representations were more task-specific (and less general) in LSTM models, but not in transformer models. In our investigation, the top layers of the (LSTM) NMT models were better for syntactic and semantic tasks. One

possible explanation for this could be that translation is more aligned with the syntactic and semantic properties than language modeling.

The development and analysis of CWRs is still ongoing. At present, NMT representations appear to be weaker than those obtained by contextualizers trained on raw texts, at least when the latter are trained on much larger amounts of data. It remains to be seen whether NMT representations can complement raw-text CWRs in certain scenarios.

10.3 On the Impact of Language Representation on Translation Output

Our methodological approach evaluates whether various linguistic properties are decodable from learned NMT representations. Our assumption was that the quality of a trained classifier can serve as a proxy to the quality of the original model, for a given task. However, it is not clear whether the NMT model really "cares" about the linguistic properties, in the sense that it relies on them for performing the translation tasks. In essence, we only provide correlational evidence, not causal evidence. This is a limitation of much of the work using classification tasks to analyze neural networks, as explained by Belinkov and Glass (2019). One avenue for addressing this question in causal terms is to define interventions: Change something in the representation and test whether and how it impacts the output translation. Bau et al. (2019b) perform such intervention experiments in NMT. They identify individual neurons that capture certain morphological properties—gender, number, and tense—and modify their activations. They evaluate how such intervention affects the output translations, finding that tense is fairly well modified, but gender and number are not as affected. Following similar ideas may be a fruitful area for further investigation of various linguistic properties and how much NMT systems depend on them when producing a translation.

10.4 Why Analyze?

There are various motivations for work on interpretability and analysis of neural network models in NLP and other domains. There are also questions concerning their necessity. Although this article does not aim to solve this debate,[22] we would like to highlight a few potential benefits of the analysis. First, several of our results may serve as guidelines for improving the quality of NMT systems and their utility for other tasks. The results on using different translation units suggest that their choice may depend on what properties one would like to capture. This may have implications for using MT systems in different languages (morphologically rich vs. poor, free vs. fixed word order) or genre (short, simple sentences vs. long, complex ones). The results on representation depth suggest that using NMT representations for contextualization may benefit from combining layers, maybe with task-wise weighting. One could also imagine performing multi-task learning of MT and other tasks, with auxiliary losses integrated in different layers. The results on multilingual systems indicate that such systems may lead to better representations, but often require greater capacity. Inspecting language representations in a zero-shot MT scenario (Johnson et al. 2017; Arivazhagan et al. 2019) may also yield new insights for improving such systems.

22 See Belinkov and Glass (2019) and references therein for considerations in the context of NLP.

10.5 Other NMT Architectures

The NMT models analyzed in this work are all based on recurrent LSTM encoder-decoder models with attention. Although this is the first successful NMT architecture, and still a dominant one, it is certainly not the only one. Other sucessful architectures include fully convolutional (Gehring et al. 2017) and fully attentional, transformer encoder-decoder models (Vaswani et al. 2017). There are also non-autoregressive models, which are promising in terms of efficiency (Gu et al. 2018). At present, NMT systems based on transformer components appear to be the most successful. Combinations of transformer and recurrent components may also be helpful (Chen et al. 2018).

The generalization of the particular results in this work to other architectures is a question of study. Recent efforts to analyze transformer-based NMT models include attempts to extract syntactic trees from self-attention weights (Mareček and Rosa 2018; Raganato and Tiedemann 2018) and evaluating representations from the transformer encoder (Raganato and Tiedemann 2018). The latter found that lower layers tend to focus on POS and shallow syntax, whereas higher layers are more focused on semantic tagging. These results are in line with our findings. However, more work is needed to understand the linguistic representational power of various NMT architectures. We expect the questions themselves, and the methods, to remain an active field of investigation with newer architectures and systems.

11. Conclusion and Future Work

In this article, we presented a comprehensive analysis of the representations learned during NMT training from the perspective of core linguistic phenomena, namely, morphology, syntax, and semantics. We evaluated the representation quality on the tasks of morphological, syntactic, and semantic tagging and using syntactic and semantic dependency labeling. Our results show that the representations learned during neural MT training learn a non-trivial amount of linguistic information. We found that different properties are represented to varying extents in different components of the NMT models. The main insights are:

- Comparing representations at different layer depths, we found that word morphology is learned at the lower layer in the LSTM encoder-decoder model, whereas non-local linguistic phenomena in syntax and semantics are better represented at the higher layers. For example, we found that higher layers are better at predicting clause-level syntactic dependencies, or second and third semantic arguments, in contrast to short-range dependencies, which do not benefit much from higher layers.
- Comparing representations with different translation units, we found that representations learned using characters perform best at capturing word morphology, and therefore provide a more viable option when translating morphologically rich languages such as Czech. They are more robust toward handling unknown and low frequency words.
- In contrast, representations learned from subword units are better at capturing syntactic and semantic information that requires learning non-local dependencies. Character-based representations, on the other hand, are poor at handling long-range dependencies and therefore inferior

when translating syntactically divergent language pairs such as German-English.

- We found morpheme-segmented units to give better representations than the ones learned using non-linguistic BPE units. The former outperformed the latter in most scenarios, even giving slightly better translation quality.
- We found that multilingual models benefit from the shared properties across different language pairs and learn richer representations compared to the bilingual model.

Future work can expand the analysis into many directions. For instance, in terms of the studied linguistic properties, moving beyond words and relations to explore phrase and sentence structures could be an interesting frontier to explore. The current study focused on NMT models based on LSTMs. Analyzing other architectures such as Transformers (Vaswani et al. 2017), which recently set a new state of the art compared to both recurrent and convolutional models (Gehring et al. 2017), would be an exciting direction to pursue.

Appendix A

A.1 Character-Based Models

The character-based models reported in this article were trained using bidirectional LSTM models only. We simply segmented words into characters and marked word boundaries. However, we did try charCNN (Kim et al. 2015; Costa-jussà and Fonollosa 2016) models in our preliminary experiments. The model is a CNN with a highway network over characters and trains an LSTM on top of it. In our results, we found the charCNN variant to perform poorly (see Table A.1), compared to the simple char-based LSTM model, both in translation quality and comparing classifier accuracy. We therefore left it out and focused on char-based LSTM models.

A.2 Effect of Target Language

Our results showed that the representations that are learned when translating into English are better for predicting POS or morphology than those learned when translating into German, which are in turn better than those learned when translating into Hebrew. The inherent difficulty in translating Arabic to (morphologically rich) Hebrew/German languages may affect the ability to learn good representations of word structure, or perhaps more data are required in the case of these languages to learn Arabic represen-

Table A.1
BLEU scores and morphological classifier accuracy across language pairs, comparing (encoders of) fully character-based LSTM and charCNN LSTM models.

	BLEU		Accuracy	
	de-en	cs-en	de-en	cs-en
char → bpe	34.9	29.0	79.3	81.7
charCNN → bpe	32.3	28.0	79.0	79.9

Table A.2
Effect of changing the target language on POS and morphological tagging with classifiers trained on the encoder side of both word-based and character-based (here: charCNN) models. The source language, for which classification is done, is always Arabic.

		ar	he	de	en
Word	POS	67.21	78.13	78.85	80.21
	Morphology	55.63	64.87	65.91	67.18
	BLEU	80.43	9.51	11.49	23.8
Char	POS	87.72	92.67	93.05	93.63
	Morphology	75.21	80.50	80.61	81.49
	BLEU	75.48	11.15	12.86	27.82

Table A.3
Impact of changing the target language on POS tagging accuracy with classifiers trained on the encoder side. Self = German/Czech in rows 1/2, respectively.

	Target language		
Source language	English	Arabic	Self
German	93.5	92.7	89.3
Czech	75.7	75.2	71.8

Table A.4
Results on syntactic and semantic tagging and labeling with representations obtained from char-based models trained with an extra layer.

	cs-en		en-de		
	Syn Dep	Sem Dep	Syn Dep	Sem Dep	Sem tags
char (layer 2)	89.3	84.3	90.3	78.9	92.3
char (layer 3)	90.2	85.2	91.1	79.6	92.7
best subword	90.3	86.3	91.4	80.4	93.2

tations of the same quality. We found these results to be consistent in other language pairs, that is, by changing the source from Arabic to German and Czech and when training character-based models instead of word-based models. See Tables A.2 and A.3 for these results.

A.3 Three Layered Character-Based Models

In order to probe whether character-based models require additional depth in the network to capture the same amount of information, we carried out further experiments training 3-layered character models for Czech-to-English and English-to-German. We extracted feature representations and trained classifiers to predict syntactic dependency labels. Our results show that using an additional layer does improve the prediction accuracy, giving the same result as subword segmentation (Morfessor) in the case of Czech-to-English, but still worse in the case of English-to-Czech (see Table A.4).

Table A.5
CCG tagging accuracy using features from the k-th encoding layer of 4-layered English-to-* NMT models trained with different target languages (German, Czech, French, and Spanish).

k	de	cs	fr	es
1	88.15	84.90	87.70	87.55
2	88.70	86.20	88.60	88.10
3	88.80	86.60	88.10	88.35
4	89.50	85.10	88.80	88.90
All	91.60	89.90	91.30	91.20

Table A.6
Statistical significance results for syntactic dependency labeling from Section 7.2. The cells above the main diagonal are for the translation direction A→ and below it are for the direction B → A. $ns = p > 0.05$, † $= p < 0.01$, ‡ $= p < 0.001$. Comparisons at empty cells are not shown.

	English-Arabic					English-Spanish					English-French				
k	0	1	2	3	4	0	1	2	3	4	0	1	2	3	4
0		‡					‡					‡			
1	‡		‡	‡	‡	‡		‡	‡	‡	‡		‡	‡	‡
2		‡		ns	‡		ns		‡	‡		†		‡	‡
3		‡	ns		‡		‡	‡		‡		‡	‡		‡
4		‡	†	ns			‡	‡	‡			‡	‡	‡	

	English-Russian					English-English				
k	0	1	2	3	4	0	1	2	3	4
0		‡					‡			
1	‡		†	‡	‡	‡		†	†	‡
2		‡		‡	‡		†		ns	‡
3		‡	ns		†		†	ns		‡
4		‡	‡	‡			‡	‡	‡	

A.4 Layer-Wise Experiments Using CCG Tags

Along with the syntactic dependency labeling task, we found higher layers to give better classifier accuracy also in the CCG tagging task. See Table A.5 for the results.

A.5 Statistical Significance Results

Table A.6 shows statistical significance results for syntactic dependency labeling experiments from Section 7.2.

Acknowledgments
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
Abdelali, lAhmed, Kareem Darwish, Nadir Durrani, and Hamdy Mubarak. 2016. Farasa: A fast and furious segmenter for Arabic. In *Proceedings of the 2016 Conference of the North American Chapter of the*

Association for Computational Linguistics: Demonstrations, pages 11−16, San Diego, CA.

Abzianidze, Lasha, 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: Volume 2, Short Papers*, pages 242–247, Valencia.

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

Aharoni, Roee and Yoav Goldberg. 2017. Towards string-to-tree neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 132–140.

Arivazhagan, Naveen, Ankur Bapna, Orhan Firat, Roee Aharoni, Melvin Johnson, and Wolfgang Macherey. 2019. The missing ingredient in zero-shot neural machine translation. *arXiv preprint arXiv:1903.07091*.

Bahdanau, Dzmitry, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Bangalore, Srinivas and Aravind K. Joshi. 1999. Supertagging: An approach to almost parsing. *Computational Linguistics*, 25(2).

Bau, D. Anthony, Yonatan Belinkov, Hassan Sajjad, Nadir Durrani, Fahim Dalvi, and James Glass. 2019a. Identifying and controlling important neurons in neural machine translation. *International Conference on Learning Representations (ICLR)*.

Bau, D. Anthony, Yonatan Belinkov, Hassan Sajjad, Nadir Durrani, Fahim Dalvi, and James Glass. 2019b. Identifying and controlling important neurons in neural machine translation. In *International Conference on Learning Representations (ICLR)*.

Bawden, Rachel, Rico Sennrich, Alexandra Birch, and Barry Haddow. 2018. Evaluating discourse phenomena in neural machine translation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1304–1313.

Belinkov, Yonatan. 2018. *On Internal Language Representations in Deep Learning: An Analysis of Machine Translation and Speech Recognition*. Ph.D. thesis, Massachusetts Institute of Technology.

Belinkov, Yonatan, 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)*, Vancouver.

Belinkov, Yonatan and James Glass. 2019. Analysis methods in neural language processing: A survey. *Transactions of the Association for Computational Linguistics (TACL)*.

Belinkov, Yonatan, 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)*.

Bentivogli, Luisa, Arianna Bisazza, Mauro Cettolo, and Marcello Federico. 2016. Neural versus phrase-based machine translation quality: A case study. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 257–267, Austin, TX.

Bisazza, Arianna and Clara Tump. 2018. The lazy encoder: A fine-grained analysis of the role of morphology in neural machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2871–2876, Association for Computational Linguistics, pages 237–265. Brussels.

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

Bojar, Ondřej, 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 (WMT17). In *Proceedings of the Second Conference on Machine Translation, Volume 2:*

Shared Task Papers, pages 169–214, Copenhagen.

Bojar, Ondřej, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, Varvara Logacheva, Christof Monz, Matteo Negri, Aurelie Neveol, Mariana Neves, Martin Popel, Matt Post, Raphael Rubino, Carolina Scarton, Lucia Specia, Marco Turchi, Karin Verspoor, and Marcos Zampieri. 2016. Findings of the 2016 conference on machine translation. In *Proceedings of the First Conference on Machine Translation*, pages 131–198, Berlin.

Bowman, Samuel R., Ellie Pavlick, Edouard Grave, Benjamin Van Durme, Alex Wang, Jan Hula, Patrick Xia, Raghavendra Pappagari, R. Thomas McCoy, Roma Patel, Najoung Kim, Ian Tenney, Yinghui Huang, Katherin Yu, Shuning Jin, and Berlin Chen. 2018. Looking for ELMo's friends: Sentence-level pretraining beyond language modeling. *ArXiv:1812.10860*.

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

Burlot, Franck and François Yvon. 2017. Evaluating the morphological competence of machine translation systems. In *Proceedings of the Second Conference on Machine Translation*, pages 43–55.

Cettolo, Mauro, 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 2016)*, Seattle, WA.

Chen, Kehai, Rui Wang, Masao Utiyama, Lemao Liu, Akihiro Tamura, Eiichiro Sumita, and Tiejun Zhao. 2017. Neural machine translation with source dependency representation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2846–2852.

Chen, Mia Xu, Orhan Firat, Ankur Bapna, Melvin Johnson, Wolfgang Macherey, George Foster, Llion Jones, Mike Schuster, Noam Shazeer, Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, Lukasz Kaiser, Zhifeng Chen, Yonghui Wu, and Macduff Hughes. 2018. The best of both worlds: Combining recent advances in neural machine translation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 76–86.

Chiang, David. 2005. A hierarchical Phrase-based model for statistical machine translation. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 263–270.

Chiang, David, Kevin Knight, and Wei Wang. 2009. 11,001 New features for statistical machine translation. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 218–226.

Chung, Junyoung, 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 (Volume 1: Long Papers)*, pages 1693–1703, Berlin.

Cinková, Silvie, Jan Hajič, Marie Mikulová, Lucie Mladová, Anja Nedolužko, Petr Pajas, Jarmila Panevová, Jiří Semeckỳ, Jana Šindlerová, Josef Toman, Zdeňka Urešová, and Zdeněk Žabokrtský. 2004. Annotation of English on the tectogrammatical level: Reference book. Technical report, ÚFAL/CKL, Prague, Czech Republic.

Cinková, Silvie, Josef Toman, Jan Hajič, Kristýna Čermáková, Václav Klimeš, Lucie Mladová, Jana Šindlerová, Kristýna Tomšů, and Zdeněk Žabokrtský. 2009. Tectogrammatical annotation of the Wall Street Journal. *The Prague Bulletin of Mathematical Linguistics*, (92):85–104.

Clark, Stephen and James R. Curran. 2004. The importance of supertagging for wide-coverage CCG parsing. In *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*.

Conneau, Alexis, German 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)*.

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

Dalvi, Fahim, Nadir Durrani, Hassan Sajjad, Yonatan Belinkov, D. Anthony Bau, and

James Glass. 2019a. What is one grain of sand in the desert? Analyzing individual neurons in deep NLP models. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence (AAAI, Oral presentation)*.

Dalvi, Fahim, 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 8th International Joint Conference on Natural Language Processing (IJCNLP)*.

Dalvi, Fahim, Avery Nortonsmith, D. Anthony Bau, Yonatan Belinkov, Hassan Sajjad, Nadir Durrani, and James Glass. 2019b. NeuroX: A toolkit for analyzing individual neurons in neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, Honolulu, HI.

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 17th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*.

Durrani, Nadir, Hassan Sajjad, Alexander Fraser, and Helmut Schmid. 2010. Hindi-to-Urdu machine translation through transliteration. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 465–474, Uppsala.

Durrani, Nadir, Hassan Sajjad, Hieu Hoang, and Philipp Koehn. 2014. Integrating an unsupervised transliteration model into statistical machine translation. In *Proceedings of the 15th Conference of the European Chapter of the ACL (EACL 2014)*, Gothenburg.

Durrani, Nadir, Helmut Schmid, and Alexander Fraser. 2011. A joint sequence translation model with integrated reordering. In *Proceedings of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT'11)*, Portland, OR.

Elman, Jeffrey L. 1991. Distributed representations, simple recurrent networks, and grammatical structure. *Machine Learning*, 7(2–3):195–225.

Eriguchi, Akiko, Kazuma Hashimoto, and Yoshimasa Tsuruoka. 2016. Tree-to-sequence attentional neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 823–833.

Galley, Michel, Jonathan Graehl, Kevin Knight, Daniel Marcu, Steve DeNeefe, Wei Wang, and Ignacio Thayer. 2006. Scalable inference and training of context-rich syntactic translation models. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 961–968.

Galley, Michel and Christopher D. Manning. 2008. A simple and effective hierarchical phrase reordering model. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 848–856, Honolulu, HI.

Ganesh, J., 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*, pages 95–102, New York, NY.

Gehring, Jonas, Michael Auli, David Grangier, Denis Yarats, and Yann N. Dauphin. 2017. Convolutional sequence to sequence learning. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 1243–1252, Sydney.

Gu, Jiatao, James Bradbury, Caiming Xiong, Victor O. K. Li, and Richard Socher. 2018. Non-autoregressive neural machine translation. In *International Conference on Learning Representations*.

Heafield, Kenneth. 2011. KenLM: Faster and smaller language model queries. In *Proceedings of the Workshop on Statistical Machine Translation (WMT'11)*, Edinburgh.

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

Hockenmaier, Julia. 2006. 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*, pages 505–512.

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

Huck, Matthias, Aleš Tamchyna, Ondřej Bojar, and Alexander Fraser. 2017.

Producing unseen morphological variants in statistical machine translation. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 369–375.

Hupkes, Dieuwke, Sara Veldhoen, and Willem Zuidema. 2017. Visualisation and 'diagnostic classifiers' reveal how recurrent and recursive neural networks process hierarchical structure. *arXiv preprint arXiv:1711.10203*.

Johnson, Melvin, Mike Schuster, Quoc Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2017. Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5339–351.

Jones, Bevan, Jacob Andreas, Daniel Bauer, Moritz Karl Hermann, and Kevin Knight. 2012. Semantics-based machine translation with hyperedge replacement grammars. In *Proceedings of COLING 2012*, pages 1359–1376.

Kádár, Akos, Grzegorz Chrupała, and Afra Alishahi. 2017. Representation of linguistic form and function in recurrent neural networks. *Computational Linguistics*, 43(4):761–780.

Karpathy, Andrej, Justin Johnson, and Fei-Fei Li. 2015. Visualizing and understanding recurrent networks. *arXiv preprint arXiv:1506.02078*.

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

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

King, Margaret and Kirsten Falkedal. 1990. Using Test Suites in Evaluation of Machine Translation Systems. In *COLING 1990 Volume 2: Papers presented to the 13th International Conference on Computational Linguistics*.

Kingma, Diederik and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Koehn, Philipp and Hieu Hoang. 2007. Factored translation models. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*.

Koehn, Philipp, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions*, pages 177–180.

Koehn, Philipp and Kevin Knight. 2003. Empirical methods for compound splitting. In *10th Conference of the European Chapter of the Association for Computational Linguistics*, pages 187–194.

Koehn, Philipp, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of the North American Chapter of the Association for Computational Linguistics on Human Language Technology (HLT-NAACL'03)*, Edmonton.

Köhn, Arne. 2015. What's in an embedding? Analyzing word embeddings through multilingual evaluation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2067–2073, Lisbon.

Komachi, Mamoru, Yuji Matsumoto, and Masaaki Nagata. 2006. Phrase reordering for statistical machine translation based on predicate-argument structure. In *International Workshop on Spoken Language Translation (IWSLT)*, pages 77–82.

Lakretz, Yair, 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)*.

Lee, Jason, 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.

Li, Junhui, Philip Resnik, and Hal Daumé III. 2013. Modeling syntactic and semantic structures in hierarchical phrase-based translation. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 540–549.

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

Lipton, Zachary C. 2016. The mythos of model interpretability. In *ICML Workshop*

on Human Interpretability in Machine Learning (WHI).

Liu, Frederick, Han Lu, and Graham Neubig. 2018. Handling homographs in neural machine translation. In *Proceedings of the 16th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Liu, Nelson F., Matt Gardner, Yonatan Belinkov, Matthew Peters, and Noah A. Smith. 2019. Linguistic knowledge and transferability of contextual representations. In *Proceedings of the 17th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*.

Luong, Minh-Thang and Christopher D. Manning. 2016. Achieving open vocabulary neural machine translation with hybrid word-character Models. *arXiv preprint arXiv:1604.00788*.

Luong, Minh-Thang, Preslav Nakov, and Min-Yen Kan. 2010. A hybrid morpheme-word representation for machine translation of morphologically rich languages. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 148–157.

Mareček, David and Rudolf Rosa. 2018. Extracting syntactic trees from transformer encoder self-attentions. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 347–349, Brussels.

McCann, Bryan, James Bradbury, Caiming Xiong, and Richard Socher. 2017. Learned in translation: Contextualized word vectors. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, Curran Associates, Inc., pages 6294–6305.

McDonald, Ryan, Kevin Lerman, and Fernando Pereira. 2006. Multilingual dependency analysis with a two-stage discriminative parser. In *Proceedings of the Tenth Conference on Computational Natural Language Learning (CoNLL-X)*, pages 216–220.

McDonald, Ryan and Joakim Nivre. 2011. Analyzing and integrating dependency parsers. *Computational Linguistics*, 37(1):197–230.

Mel'čuk, Igor Aleksandrovič. 1988. *Dependency Syntax: Theory and Practice*. SUNY Press.

Nadejde, Maria, Siva Reddy, Rico Sennrich, Tomasz Dwojak, Marcin Junczys-Dowmunt, Philipp Koehn, and Alexandra Birch. 2017. Predicting target language CCG supertags improves neural machine translation. In *Proceedings of the Second Conference on Machine Translation*, pages 68–79.

Nakov, Preslav and Jörg Tiedemann. 2012. Combining word-level and character-level models for machine translation between closely-related languages. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 301–305, Jeju.

Neubig, Graham and Kevin Duh. 2014. On the elements of an accurate tree-to-string machine translation system. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 143–149.

Nguyen, Dat Quoc, 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 Demonstrations at the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 17–20, Gothenburg.

Nivre, Joakim. 2005. Dependency Grammar and Dependency Parsing., Technical Report MSI 015133, Växjö University, School of Mathematics and Systems Engineering.

Nivre, Joakim, Željko Agić, Lars Ahrenberg, Maria Jesus Aranzabe, Masayuki Asahara, Aitziber Atutxa, Miguel Ballesteros et al. 2017. Universal Dependencies 2.0. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics, Charles University.

Oepen, Stephan, Marco Kuhlmann, Yusuke Miyao, Daniel Zeman, Silvie Cinkova, Dan Flickinger, Jan Hajic, and Zdenka Uresova. 2015. Semeval 2015 task 18: Broad-coverage semantic dependency parsing. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 915–926.

Oepen, Stephan, Marco Kuhlmann, Yusuke Miyao, Daniel Zeman, Dan Flickinger, Jan Hajic, Angelina Ivanova, and Yi Zhang. 2014. Semeval 2014 Task 8: Broad-coverage semantic dependency parsing. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 63–72.

Padó, Sebastian. 2006. User's guide to `sigf`: Significance testing by approximate randomisation. `https://www.nlpado.de/~sebastian/software/sigf.shtml`.

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*.

Pasha, Arfath, Mohamed Al-Badrashiny, Mona Diab, Ahmed El Kholy, Ramy Eskander, Nizar Habash, Manoj Pooleery, Owen Rambow, and Ryan Roth. 2014. MADAMIRA: A fast, comprehensive tool for morphological analysis and disambiguation of Arabic. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 1094–1101, Reykjavik.

Peters, Matthew, Sebastian Ruder, and Noah A. Smith. 2019. To tune or not to tune? Adapting pretrained representations to diverse tasks. *arXiv preprint arXiv:1903.05987*.

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

Peters, Matthew E., Mark Neumann, Luke Zettlemoyer, and Wen-tau Yih. 2018b. Dissecting contextual word embeddings: Architecture and representation. In *Proceedings of EMNLP*.

Pinnis, Mārcis, Rihards Krišlauks, Toms Miks, Daiga Deksne, and Valters Šics. 2017. Tilde's machine translation systems for WMT 2017. In *Proceedings of the Second Conference on Machine Translation, Volume 2: Shared Task Papers*, pages 374–381, Copenhagen.

Qian, Peng, 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*, pages 826–835, Austin.

Qian, Peng, 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 (Volume 1: Long Papers)*, pages 1478–1488, Berlin.

Radford, Alec, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training. Technical report, OpenAI. `https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/language-unsupervised/language_understanding_paper.pdf`.

Radford, Alec, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. Technical report, OpenAI. `https://d4mucfpksywv.cloudfront.net/better-language-models/language_models_are_unsupervised_multitask_learners.pdf`.

Raganato, Alessandro and Jörg Tiedemann. 2018. An analysis of encoder representations in transformer-based machine translation. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 287–297.

Renduchintala, Adithya, Pamela Shapiro, Kevin Duh, and Philipp Koehn. 2018. Character-aware decoder for neural machine translation. *arXiv preprint arXiv:1809.02223*.

Rios Gonzales, Annette, Laura Mascarell, and Rico Sennrich. 2017. Improving word sense disambiguation in neural machine translation with sense embeddings. In *Proceedings of the Second Conference on Machine Translation*, pages 11–19.

Schmid, Helmut. 1994. Part-of-speech tagging with neural networks. In *Proceedings of the 15th International Conference on Computational Linguistics (COLING 1994)*, pages 172–176, Kyoto.

Sennrich, Rico. 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: Volume 2, Short Papers*, pages 376–382, Valencia.

Sennrich, Rico, Alexandra Birch, Anna Currey, Ulrich Germann, Barry Haddow, Kenneth Heafield, Antonio Valerio Miceli Barone, and Philip Williams. 2017. The University of Edinburgh's neural MT systems for WMT17. In *Proceedings of the Second Conference on Machine Translation, Volume 2: Shared Task Papers*, pages 389–399, Copenhagen.

Sennrich, Rico, 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 (Volume 1: Long Papers)*, pages 1715–1725, Berlin.

Shapiro, Pamela and Kevin Duh. 2018. BPE and CharCNNs for translation of morphology: A cross-lingual comparison and analysis. *arXiv preprint arXiv:1809.01301*.

Shen, Libin, Jinxi Xu, and Ralph Weischedel. 2010. String-to-dependency statistical machine translation. *Computational Linguistics*, 36(4):649–671.

Shi, Xing, 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*, pages 1526–1534 Austin, TX.

Smit, Peter, Sami Virpioja, Stig-Arne Grönroos, and Mikko Kurimo. 2014. 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*, pages 21–24, Gothenburg.

Stahlberg, Felix, Eva Hasler, Aurelien Waite, and Bill Byrne. 2016. Syntactically guided neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 299–305.

Steedman, Mark and Jason Baldridge. 2011. *Combinatory Categorial Grammar*. chapter 5. John Wiley and Sons, Ltd.

Sutskever, Ilya, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, pages 3104–3112.

Tenney, Ian, Patrick Xia, Berlin Chen, Alex Wang, Adam Poliak, R. Thomas McCoy, Najoung Kim, Benjamin Van Durme, Samuel R. Bowman, Dipanjan Das, and Ellie Pavlick. 2019. What do you learn from context? Probing for sentence structure in contextualized word representations. In *Proceedings of ICLR*.

Toral, Antonio and Víctor M. Sánchez-Cartagena. 2017. A multifaceted evaluation of neural versus phrase-based machine translation for 9 language directions. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 1063–1073, Valencia.

Tran, Ke, Arianna Bisazza, and Christof Monz. 2018. The importance of being recurrent for modeling hierarchical structure. *arXiv preprint arXiv:1803.03585*.

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, Curran Associates, Inc., pages 5998–6008.

Vylomova, Ekaterina, Trevor Cohn, Xuanli He, and Gholamreza Haffari. 2016. Word representation models for morphologically rich languages in neural machine translation. *arXiv preprint arXiv:1606.04217*.

Wang, Yu Hsuan, Cheng-Tao Chung, and Hung-yi Lee. 2017. Gate activation signal analysis for gated recurrent neural networks and its correlation with phoneme boundaries. In *Interspeech 2017*.

Weaver, Warren. 1955. Translation. *Machine Translation of Languages*, 14:15–23.

Williams, Philip, Rico Sennrich, Matt Post, and Philipp Koehn. 2016. Syntax-based statistical machine translation. *Synthesis Lectures on Human Language Technologies*, 9(4):1–208.

Wu, Shuangzhi, Dongdong Zhang, Nan Yang, Mu Li, and Ming Zhou. 2017. Sequence-to-dependency neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 698–707.

Wu, Xianchao, Katsuhito Sudoh, Kevin Duh, Hajime Tsukada, and Masaaki Nagata. 2011. Extracting pre-ordering rules from predicate-argument structures. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pages 29–37.

Wu, Yonghui, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, et al. 2016a. Google's neural machine translation system: Bridging the gap between human and machine translation. *CoRR*, abs/1609.08144.

Wu, Yonghui, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016b. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

Wu, Zhizheng and Simon King. 2016. Investigating gated recurrent networks for speech synthesis. In *2016 IEEE International*

Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 5140–5144.

Xiong, Deyi, Min Zhang, and Haizhou Li. 2012. Modeling the translation of predicate-argument structure for SMT. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 902–911.

Xu, Wenduan, Michael Auli, and Stephen Clark. 2015. CCG supertagging with a recurrent neural network. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 250–255, Beijing.

Yamada, Kenji and Kevin Knight. 2002. A decoder for syntax-based statistical MT. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*.

Zhang, Kelly W. and Samuel R. Bowman. 2018. Language modeling teaches you more syntax than translation does: Lessons learned through auxiliary task analysis. In *Proceedings of BlackboxNLP*.

Zhang, Min, Hongfei Jiang, AiTi Aw, Jun Sun, Sheng Li, and Chew Lim Tan. 2007. A tree-to-tree alignment-based model for statistical machine translation. *MT-Summit-07*, pages 535–542.

Zhou, Jie, Ying Cao, Xuguang Wang, Peng Li, and Wei Xu. 2016. Deep recurrent models with fast-forward connections for neural machine translation. *Transactions of the Association for Computational Linguistics*, 4:371–383.

Ziemski, Michal, Marcin Junczys-Dowmunt, and Bruno Pouliquen. 2016. The United Nations Parallel Corpus v1.0. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris.